

# Pytania na Egzamin Kierunkowy po I stopniu studiów stacjonarnych

oraz

## Sprawdzian Kwalifikacyjny na II stopień studiów stacjonarnych dla kierunku Informatyka stosowana

### 1 Języki i metody programowania

1. IT1A\_W03,IT1A\_U03,IT1A\_U07 W jaki sposób można obliczyć długość tekstu przekazanego jako argument w poniższej funkcji?

```
1 void foo(const char*txt){
2     ...
3 }
```

Przykładowo odp.) sizeof(txt)

2. IT1A\_W03,IT1A\_U03,IT1A\_U07 Co możesz powiedzieć o poniższej deklaracji?

```
1 int t[10]={1,2,[4]=1}
```

Przykładowo odp.) Zgodnie ze standardem C99 jest ona niepoprawna.

3. IT1A\_W03,IT1A\_U03,IT1A\_U07 W jaki sposób obliczyć długość tablicy w funkcji foo()?

```
1 void foo(double t[]){
2     // dlugosc tablicy t?
3 }
```

Przykładowo odp.) Jako

```
1     sizeof(t)/sizeof(t[0])
```

4. IT1A\_W03,IT1A\_U03,IT1A\_U07 Która z implementacji funkcji zwracającej tablicę jest poprawna?

Przykładowo odp.)

```
1
2 int * getTable ()
3 {
4     int tab[10];
5     return tab;
6 }
```

5. IT1A\_W03,IT1A\_U03,IT1A\_U07 Zakładając, że wielkość typu char to jeden bajt, short to dwa bajty, a double to osiem bajtów, jaka jest wartość wyrażenia sizeof(x), gdzie x jest zmienną poniższego typu strukturalnego, dla standardowych ustawień kompilatora 32-bitowego?

```
1 struct {
2     char c;
3     short i;
4     double d;
5 } x;
```

Przykładowo odp.) 8+8+8=24

**6. IT1A\_W03,IT1A\_U03,IT1A\_U07** Przeanalizuj poniższą deklarację? Jakie wartości wyrażeń, w których występują wskaźniki p1 i p2 zostaną wydrukowane? (Założ, że używasz 32-bitowego kompilatora.)

```
1 int t[10];
2 int *p1=&t[0];
3 int *p2=&t[8];
```

Przykładowo odp.) Poniższa instrukcja wypisze 8

```
1 printf("%d\n", p2-p1);
```

**7. IT1A\_W03,IT1A\_U03,IT1A\_U07** Przeanalizuj poniższą deklarację w języku C:

```
1 int (*x)(int, int);
```

Przykładowo odp.) Deklaracja jest niezgodna ze składnią języka.

**8. IT1A\_W03,IT1A\_U03,IT1A\_U07** Które stwierdzenia dotyczące operatorów w języku C/C++ są poprawne:

Przykładowo odp.) Operatory addytywne mają mniejszy priorytet niż multiplikatywne.

**9. IT1A\_W03,IT1A\_U03,IT1A\_U07** Które stwierdzenia dotyczące modyfikatora static w języku C/C++ są poprawne:

Przykładowo odp.) Pole static klasy C++ wymaga zadeklarowania zmiennej w analogiczny sposób, jak deklaruje się zmienne globalne wewnątrz modułów.

**10. IT1A\_W03,IT1A\_U03,IT1A\_U07** Dzięki konwencji wywołania funkcji w języku C znanej jako `__cdecl` możliwa jest implementacja funkcji o zmiennej liczbie argumentów, jak `printf()`. Które stwierdzenia charakteryzujące funkcje typu `__cdecl` są prawdziwe?

Przykładowo odp.) W wygenerowanym kodzie wywołania funkcji argumenty umieszczane są na stosie od końca. Dzięki temu na szczycie stosu jest jej pierwszy argument i analizując jego zawartość można określić spodziewaną liczbę argumentów wywołania.

**11. IT1A\_W03,IT1A\_U03,IT1A\_U07** W jaki sposób przekazywany jest parametr będący tablicą do funkcji w języku C, np.:

```
1 int main(int argc, char* argv[]){
2     //...
3 }
```

Przykładowo odp.) Na stosie umieszczany jest adres pierwszego elementu tablicy.

**12. IT1A\_W03,IT1A\_U03,IT1A\_U07** Które stwierdzenia odnoszące się do przydziału pamięci dla zmiennych w językach C i C++ są prawdziwe?

Przykładowo odp.) Przydział pamięci dla zmiennych deklarowanych z modyfikatorem `static` następuje przed rozpoczęciem wykonania programu.

**13. IT1A\_W04,IT1A\_U04,IT1A\_U07,IT1A\_U21**

Które ze stwierdzeń odnoszących się do referencji w języku C++ są poprawne?

Przykładowo odp.) Poniższy kod w poprawny sposób przenosi referencję z obiektu a na b:

```
1 A a;
2 A b;
3 A&r = a;
4 r=b;
```

**14. IT1A\_W04,IT1A\_U04,IT1A\_U07,IT1A\_U21** Jeżeli podczas wykonania instrukcji w C++:

```
1 A* ptr = new A();
```

wygenerowany został wyjątek, jego przyczyna może być następująca:

Przykładowo odp.) Klasa A jest klasą abstrakcyjną i nie można utworzyć obiektu tej klasy.

**15. IT1A\_W04,IT1A\_U04,IT1A\_U07,IT1A\_U21** Przeanalizuj fragment kodu w języku C++, w którym pojawia się wywołanie operatora «

```
1 A a;  
2 std::cout<<a;
```

Która z podanych implementacji operatora « jest poprawna (przykładowy kod zostanie skompilowany i wykonany)?

Przykładowo odp.) Jako funkcja niezwracająca wartości

```
1 void operator<<( std::ostream&os , const A&a ) { ... }
```

**16. IT1A\_W04,IT1A\_U04,IT1A\_U07,IT1A\_U21** Zdefiniowano szablon (wzorzec) funkcji

```
1 template<class T>  
2 T suma(T*table , int size )  
3 {  
4     T t=T();  
5     for( int i=0;i<size ; i++) t+=table [ i ];  
6     return t;  
7 }
```

Proces instancjacji szablonu polega na zastąpieniu typów i zmiennych będących parametrami szablonu konkretnymi typami i wartościami, a następnie generacji kodu wynikowego. Jakie założenia musi spełniać typ T, aby instancjacja szablonu była możliwa?

Przykładowo odp.) Typem T użytym podczas instancjacji szablonu może być typ wbudowany (int, char, float, double)

**17. IT1A\_W04,IT1A\_U04,IT1A\_U07,IT1A\_U21** Klasa B przechowuje wskaźniki do obiektów klasy A w kontenerze vector standardowej biblioteki C++ (STL)

```
1 class A { ... };  
2 class B { public :  
3     std::vector<A*> v ;  
4     void add(A&a) { v.push_back(new A(a)); }  
5     ~B();  
6 };
```

Która z implementacji destruktora jest poprawna (kompiluje się, nie prowadzi do błędów wykonania lub wycieków pamięci)?

Przykładowo odp.)

```
1 B::~~B() { v.clear(); }
```

**18. IT1A\_W04,IT1A\_U04,IT1A\_U07,IT1A\_U21** Szablon set<T> zdefiniowany w standardowej bibliotece C++ (STL) przechowuje elementy w drzewiastych strukturach danych. Który z przedstawionych typów danych może być zastosowany jako parametr instancjacji szablonu set<T>?

Przykładowo odp.) Typ string

**19. IT1A\_W04,IT1A\_U04,IT1A\_U07,IT1A\_U21** Które ze stwierdzeń odnoszących się do konstruktorów kopiujących i operatorów przypisania w języku C++ są poprawne?

Przykładowo odp.) Standardowa implementacja (automatycznie wygenerowana przez kompilator) konstruktora kopiującego C++ kopiuje kolejne bajty składające się na pamięć obiektu.

**20. IT1A\_W04,IT1A\_U04,IT1A\_U07,IT1A\_U21** Implementacja przeciążonych operatorów C++ powinna odzwierciedlać semantykę operacji na typach wbudowanych. Biorąc pod uwagę to wymaganie, które z implementacji operatorów dla klasy X zadeklarowanej poniżej jest poprawna?

```
1 class X
2 {
3 friend X&operator+=(X&a , const X&b );
4     int x;
5 public :
6     X(int _x=0):x(_x){}
7     X&operator+(const X&o);
8     X&operator++(int );
9     X&operator --=(const X&o);
10 };
```

Przykładowo odp.)

```
1 X&X::operator --=(const X&o){X ret=*this; ret.x-=o.x; return ret;}
```

**21. IT1A\_W04,IT1A\_U04,IT1A\_U07,IT1A\_U21** W języku C++ dostęp do informacji o typie obiektu w trakcie wykonania programu umożliwiają następujące operatory:

Przykładowo odp.) `dynamic_cast`

**22. IT1A\_W04,IT1A\_U04,IT1A\_U07,IT1A\_U21** Zadeklarowano dwie klasy w następujący sposób:

```
1 class A{
2 public :
3     virtual void f(){ printf("VA ");}
4     void g(){ printf("A ");}
5 };
6
7 class B:public A{
8 public :
9     void f(){ printf("VB ");}
10    void g(){ printf("B ");}
11 };
```

oraz utworzono dwa obiekty:

```
1 A* a1 = new A();
2 A* a2 = new B();
```

Przykładowo odp.) Wywołanie

```
1 a1->g ();
2 a2->g ();
```

spowoduje wypisanie A A

## 2 Wstęp do systemów uniksowych

23. IT1A\_W09 Podstawowa architektura Unixa obejmuje:

Przykładowo odp.) podsystem procesów

24. IT1A\_W09 W systemie plików Unix:

Przykładowo odp.) względną ścieżką dostępu określa położenie pliku względem katalogu bieżącego

25. IT1A\_U09 Prawo dostępu do pliku 453 pozwala

Przykładowo odp.) wszystkim czytać plik

26. IT1A\_U09 W systemie plików

Przykładowo odp.) pliki które zmieniają się często są w katalogu */var*

27. IT1A\_W09 Które z poniższych stwierdzeń są prawdziwe?

Przykładowo odp.) każde konto musi należeć do co najmniej jednej grupy

28. IT1A\_W09 Przy zarządzaniu systemami plików

Przykładowo odp.) system plików sprawdzamy przez **checkfs**

29. IT1A\_U09 W trakcie startu systemu Unix

Przykładowo odp.) rozróżnia się 7 podstawowych poziomów pracy (0-6)

30. IT1A\_W09 Procesy w systemie Unix

Przykładowo odp.) są identyfikowane przez PID

31. IT1A\_U09 Przykłady komunikacji międzyprocesowej w Unixie to

Przykładowo odp.) potoki (pipe)

32. IT1A\_W09 Rejestrowanie zdarzeń w Unixie:

Przykładowo odp.) służy do niego pakiet *Logger*

33. IT1A\_U10,IT1A\_W10 Przy konfiguracji komunikacji sieciowej w Unix:

Przykładowo odp.) polecenie **route** służy do konfiguracji routingu

34. IT1A\_U09 Pliki konfiguracyjne powłoki Bash w systemie Unix:

Przykładowo odp.) */etc/logout* – jest wczytywany przy zamykaniu każdej powłoki

35. IT1A\_U09 W wyniku którego z poniższych poleceń członkowie grupy, do której należy plik, stracą prawo do jego modyfikacji

Przykładowo odp.) `chmod 731 plik`

36. IT1A\_U09 Które z poniższych stwierdzeń dotyczących sygnałów przesyłanych do procesów systemie Unix są poprawne

Przykładowo odp.) sygnał SIGSEGV jest związany z naruszeniem ochrony pamięci

37. IT1A\_W10,IT1A\_U10 Przy konfiguracji obsługi sieci w Unixie:

Przykładowo odp.) plik */etc/resolv.conf* dotyczy konfiguracji DNS

### 3 Algorytmy i struktury danych

**38. IT1A\_W07** Które stwierdzenia spośród poniższych są prawdziwe

Przykładowo odp.) Pesymistyczna i oczekiwana złożoność obliczeniowa są sobie równe dla sortowania quick-sort.

**39. IT1A\_W07, IT1A\_U07** Dany jest ustalony ciąg  $n$  macierzy o tak dobranych rozmiarach, że macierze te możemy wymnożyć.

Przykładowo odp.) Algorytm optymalnego nawiasowania w problemie mnożenia  $n$  macierzy musi mieć złożoność wykładniczą ze względu na wykładniczą złożoność algorytmu rekurencyjnego obliczającego liczby Catalana.

**40. IT1A\_W01, IT1A\_W03, IT1A\_W07** W drzewie binarnym przeszukiwanie zgodnie z porządkiem inorder ma postać

Przykładowo odp.) zbadaj wg kolejności: lewe poddrzewo, wierzchołek, prawe poddrzewo

**41. IT1A\_U07** Zadanie o rozmiarze  $n$ , realizowane pewnym algorytmem o złożoności  $f(n)$ , zostało sprowadzone do dwóch podzadań o rozmiarze  $\frac{n}{2}$  każde oraz do  $n$  działań o stałym czasie wykonania, zapewniających rozbięcie i scalenie zadania. Złożoność  $f(n)$  wynosi:

Przykładowo odp.)  $f(n) = O(n \cdot \log n)$ ,

**42. IT1A\_W01, IT1A\_W03, IT1A\_W07** Dany jest graf skierowany  $G=(V,E)$ , gdzie  $V=\{1,2,3,4,5,6\}$ ,  $E=\{(1,2), (1,3), (2,4), (2,5), (4,5), (5,1), (3,5), (3,6)\}$ . Jeśli graf  $G$  przeszukujemy w głąb poczynając od wierzchołka 1 to

Przykładowo odp.) krawędź  $(1,2)$  jest krawędzią drzewową;

**43. IT1A\_W01, IT1A\_W03, IT1A\_W07** Które stwierdzenia spośród poniższych są prawdziwe

Przykładowo odp.) Algorytm sortowania topologicznego bazuje na przeszukiwaniu grafu wszerz.

**44. IT1A\_W03, IT1A\_W07, IT1A\_U07** Dana jest procedura: Proc( $n$ ) { if(warunek( $x$ )) then { A( $x$ ); Proc( $f(n)$ ); B( $x$ ) } else C( $x$ ) }. Przyjmijmy konwencję, że np. zapis AAABCC oznacza trzykrotne wykonanie instrukcji A, po czym następuje wykonanie instrukcji B a następnie dwukrotne wykonanie instrukcji C. Następujące sekwencje instrukcji mogą być wynikami wywołania powyższej procedury:

Przykładowo odp.) AACBB;

**45. IT1A\_W01, IT1A\_W03, IT1A\_W07** Graf  $G = (V, E)$  jest drzewem BST, przy czym  $V = \{15, 21, 23, 29, 31, 38, 40, 61, 96, 98\}$ ,

$E = \{(21, 15), (21, 23), (29, 21), (29, 31), (38, 29), (38, 96), (96, 40), (96, 98), (40, 61)\}$ .

Przykładowo odp.) W wyniku przeszukiwania postorder wierzchołki zostaną odwiedzone w następującej kolejności:

15, 23, 21, 29, 31, 61, 40, 98, 96, 38.

**46. IT1A\_W01, IT1A\_W03, IT1A\_W07** Niech  $p = (x_1, y_1)$ ,  $q = (x_2, y_2)$ ,  $r = (x_3, y_3)$  oraz niech  $det(p, q, r)$

oznacza wyznacznik macierzy  $\begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{bmatrix}$ .

Przykładowo odp.) Jeśli  $det(p, q, r) > 0$  to punkt  $r$  leży po lewej stronie wektora  $\vec{pq}$ .

**47. IT1A\_W01, IT1A\_W03, IT1A\_W07** Danych jest  $n$  punktów wyznaczających wielobok o  $n$  bokach.

Przykładowo odp.) Istnieje algorytm o złożoności  $O(n)$  sprawdzający, czy zadany punkt należy do wnętrza wieloboku.

**48. IT1A\_W01, IT1A\_W03, IT1A\_W07** Graf dynamiczny, którego maksymalnej liczby wierzchołków i krawędzi w trakcie wykonywania algorytmu nie potrafimy z góry oszacować powinien być reprezentowany jako

Przykładowo odp.) lista list

**49. IT1A\_W07** Dla problemu komiwojagera algorytm pozwalający wyznaczyć rozwiązanie optymalne:

Przykładowo odp.) istnieje i ma złożoność rzędu silnia

**50. IT1A\_U07** Głębokość rekurencji dla ciągu Fibonacciego zaimplementowanego rekurencyjnie zgodnie z arytmetyczną definicją rekurencyjną wynosi

Przykładowo odp.)  $O(n^4)$

**51. IT1A\_W01, IT1A\_W03, IT1A\_W07** Kursorowa implementacja listy jest strukturą

Przykładowo odp.) tablicową

**52. IT1A\_W01, IT1A\_W03, IT1A\_W07** Problem chińskiego listonosza polega na

Przykładowo odp.) znalezieniu najkrótszej drogi zamkniętej zawierającej wszystkie wierzchołki grafu

## 4 Podstawy grafiki komputerowej

**53. IT1A\_W14** Na czym polega rendering obiektu w grafice?

Przykładowo odp.) Na optymalizacji struktury obiektu.

**54. IT1A\_W14** Proszę podać która wersja etapów w tzw. „graphics pipeline” jest poprawna.

Przykładowo odp.) Modeling transformation; Viewing transformation; Projection transformation; Clipping; Per-vertex lighting; Texturing; Scan conversion or rasterization; Display

**55. IT1A\_W14** Proszę podać jakie są podstawowe (dziś) typy grafiki komputerowej?

Przykładowo odp.) Oparta na programowaniu wizualnym, oparta na językach programowania, oparta na procesorach graficznych.

**56. IT1A\_W14** Czym różni się OpenGL od Direct3D?

Przykładowo odp.) Nie ma istotnej różnicy.

**57. IT1A\_W14** Jakie są 3 podstawowe transformacje w grafice komputerowej i jaki aparat matematyczny jest używany do liczenia transformacji obiektów na scenie?

Przykładowo odp.) Skalowanie, rzutowanie, redukcja. Liczone są przy pomocy funkcji Hermite’a.

**58. IT1A\_W14** Co to jest Ray Tracing?

Przykładowo odp.) Jest to typ karty graficznej.

**59. IT1A\_W14** Jakim skrótem oznacza się powszechnie procesor graficzny?

Przykładowo odp.) CPU

**60. IT1A\_W14** Co to jest fraktal?

Przykładowo odp.) Jest to element funkcjonalny procesora graficznego.

**61. IT1A\_W14** Co oznacza NURBS?

Przykładowo odp.) Typ przetwarzania obrazu w grafice.

**62. IT1A\_W14** Co to jest Z-buforowanie?

Przykładowo odp.) Jest to sprzętowy algorytm liczenia które fragmenty sceny są widoczne.



## 5 Programowanie obiektowe

63. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21 Jakiego są typ i wartość wyrażenia `2+"2.68"`

Przykładowo odp.) Wyrażenie jest niezgodne ze składnią języka.

64. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21 Aby sprawdzić, czy dwa obiekty typu `String` mają taką samą zawartość można

Przykładowo odp.) Użyć metody `compareTo()`

65. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21 Który z poniższych fragmentów kodu sprawdza, czy obiekt wskazywany przez referencję `xyz` należy do klasy `XYZ`

Przykładowo odp.)

```
1 if (xyz.dynamicCastTo(XYZ.class) != null )
```

66. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21 Tablica jest zadeklarowana jako:

```
1 int tab[] = new int[]{3,2,1,0};
```

Który z fragmentów kodu poprawnie wypisze jej elementy

Przykładowo odp.)

```
1 for (int i=0; i<tab.size(); i++)  
2 System.out.println(i+" ");
```

67. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21 Przeanalizuj poniższy kod:

```
1 loop: for (int i=0; i<3; i++){  
2     for (int j=0; j<5; j++){  
3         System.out.print(i+j);  
4         if (j==1) break loop;  
5     }  
6 }
```

Co zostanie wypisane:

Przykładowo odp.) 011223

68. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21 Które zdanie opisujące własności klas jest **prawdziwe**

Przykładowo odp.) Dla każdej klasy w języku Java możliwe jest zdefiniowanie klasy potomnej

69. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21 Które zdanie opisujące własności klas w języku Java jest **prawdziwe**

Przykładowo odp.) Aby zaznaczyć, że klasa dziedziczy po kilku klasach należy podać ich listę po słowie kluczowym `extends`, np. `class D extends A, B, C {}`

70. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21 Które zdanie dotyczące trybów dostępu w języku Java jest **prawdziwe**

Przykładowo odp.) Dostęp do składowych `protected` jest możliwy wyłącznie z poziomu klas potomnych danej klasy

71. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21 Która kombinacja modyfikatorów metod jest dopuszczalna

Przykładowo odp.) `static synchronized`

72. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21 Które ze stwierdzeń jest **prawdziwe**:

Przykładowo odp.) Aby wykonanie metody `clone()` powiodło się, należy zadeklarować, że klasa implementuje interfejs `Cloneable`

**73. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21** W jaki sposób usuwane są obiekty w języku Java?

Przykładowo odp.) Usuwa się za pomocą operatora `delete`

**74. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21** Które z poniższych stwierdzeń odnoszących się do konstruktorów klas są **prawdziwe**:

Przykładowo odp.) Konstruktor jest wywołany podczas tworzenia obiektu za pomocą operatora `new`

**75. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21** Które ze stwierdzeń odnoszących się do wyjątków w języku Java są **prawdziwe**?

Przykładowo odp.) Deklarując klasy wyjątków należy wybrać jako klasę bazową `RuntimeException`. Za wyborem przemawia ich większa efektywność, ponieważ są one przetwarzane w pierwszej kolejności, przed pozostałymi wyjątkami.

**76. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21** Które z poniższych stwierdzeń odnoszących się do typów generycznych w języku Java są **prawdziwe**?

Przykładowo odp.) Nie jest możliwe utworzenie tablic typów parametryzowanych

**77. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21** Które z poniższych stwierdzeń odnoszące się do klas wewnętrznych i zagnieżdżonych w języku Java są **prawdziwe**

Przykładowo odp.) W metodach klasy zagnieżdżonej (zadeklarowanej z modyfikatorem `static`) jest dostępna referencja `Outer.this`, gdzie `Outer` jest nazwą klasy zewnętrznej

**78. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21** Które z poniższych stwierdzeń odnoszących się do interfejsów w języku Java są **prawdziwe**:

Przykładowo odp.) Każda metoda zadeklarowana wewnątrz interfejsu jest publiczna

**79. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21** Które stwierdzenie odnoszące się do wątków w języku Java jest **prawdziwe**

Przykładowo odp.) W kontekście używanym podczas przełączania wątków zapisane są rejestry, stos, bloki pamięci, pliki i kod

**80. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21** Które stwierdzenie odnoszące się do wątków w języku Java jest **prawdziwe**

Przykładowo odp.) Po wywołaniu `sleep(1000)` wątek zostanie uśpiony i po dokładnie 1000 milisekundach wznowi działanie.

**81. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21** Które stwierdzenia odnoszące się do monitorów w języku Java są **prawdziwe**:

Przykładowo odp.) Wątek będący właścicielem monitora może wywoływać inne metody synchroniczne.

**82. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21** Które z poniższych stwierdzeń odnoszących się do rozwiązań stosowanych w bibliotece AWT jest **prawdziwe**:

Przykładowo odp.) Każdy z komponentów posiada swój odpowiednik w postaci obiektu platformy (ang. *peer*)

**83. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21** Które z poniższych stwierdzeń odnoszących się do obsługi zdarzeń w bibliotece AWT są **prawdziwe**:

Przykładowo odp.) Komponent będący źródłem zdarzeń może mieć wielu zarejestrowanych odbiorców (`Listener`)

**84. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21** Które z poniższych stwierdzeń odnoszących się do biblioteki Swing są **prawdziwe**:

Przykładowo odp.) Lekkie komponenty Swing nie są skojarzone z obiektem platformy (oknem, kontrolką) i są zaimplementowane w języku Java

**85. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21** Modelem dla komponentu Swing jest:

Przykładowo odp.) Zbiór danych podlegających wizualizacji, najczęściej implementowany w postaci odpowiedniej klasy

**86. IT1A\_W04,IT1A\_W06,IT1A\_U04,IT1A\_U07,IT1A\_U21** W terminologii Swing *Renderer* to:

Przykładowo odp.) Funkcja odpowiedzialna za serializację zawartości kontenera w odpowiednim formacie, np. XML

## 6 Architektury komputerów

**87. IT1A\_W08,IT1A\_U08** Do czego służy jednostka sterująca

Przykładowo odp.) nie licząc licznika cykli jest to zawsze układ sekwencyjny

**88. IT1A\_W08,IT1A\_U08** ALU

Przykładowo odp.) musi być układem sekwencyjnym

**89. IT1A\_W08,IT1A\_U08** Korzystając z układu FPGA można wykonać

Przykładowo odp.) na przykład dowolny układ kombinacyjny, ograniczony jedynie wielkością struktury FPGA

**90. IT1A\_W08,IT1A\_U08** Układ kombinacyjny to

Przykładowo odp.) jest to układ logiczny nie pamiętający stanów poprzednich

**91. IT1A\_W08,IT1A\_U08** Układ sekwencyjny to

Przykładowo odp.) w skład jego mogą wchodzić bramki logiczne w połączeniu z przerzutnikami JK

**92. IT1A\_W08,IT1A\_U08** Pamięć ram

Przykładowo odp.) posiada wejścia adresowe, wejścia sterujące oraz wejście/wyjście danych

**93. IT1A\_W08,IT1A\_U08** Pamięć ram dwuportowa

Przykładowo odp.) można ją wykorzystać wyłącznie w procesorach wielordzeniowych

**94. IT1A\_W08,IT1A\_U08** Licznik

Przykładowo odp.) asynchroniczny możemy wykonać z przerzutników JK

**95. IT1A\_W08,IT1A\_U08** Procesor

Przykładowo odp.) możemy wykonać przy użyciu FPGA ale tylko jednodzeniowy

**96. IT1A\_W08,IT1A\_U08** Lista rozkazów procesora

Przykładowo odp.) w skład listy rozkazów zawsze wchodzi mnożenie

**97. IT1A\_W08,IT1A\_U08** Karta graficzna

Przykładowo odp.) przy użyciu FPGA nie można zbudować karty graficznej ze sprzętowym wspomaganie OpenGL

**98. IT1A\_W08,IT1A\_U08** Klawiatura

Przykładowo odp.) transmisja z użyciem ps2 jest transmisją asynchroniczną jednokierunkową

**99. IT1A\_W08,IT1A\_U08** Interfejs rs232

Przykładowo odp.) wykorzystuje transmisję szeregową

**100. IT1A\_W08,IT1A\_U08** Licznik rozkazów

Przykładowo odp.) jest to licznik z wejściem równoległym wykorzystywanym wyłącznie przy skokach bezwarunkowych

**101. IT1A\_W08,IT1A\_U08** Rozkaz skoku bezwarunkowego procesora

Przykładowo odp.) nie wpływa na stan licznika rozkazów procesora

**102. IT1A\_W08,IT1A\_U08** Rozkaz skoku warunkowego procesora

Przykładowo odp.) żaden z powyższych

**103. IT1A\_W08,IT1A\_U08** Rozkaz procesora wykonujący dodanie dwóch liczb

Przykładowo odp.) żaden z powyższych

**104. IT1A\_W08,IT1A\_U08** Licznik cykli procesora

Przykładowo odp.) każdy rozkaz konkretnego procesora trwa dokładnie tyle samo cykli

- 105. IT1A\_W08,IT1A\_U08** W procesorze wykorzystującym przetwarzanie potokowe  
Przykładowo odp.) rozpoczęcie wykonania rozkazu może nastąpić dopiero po zakończeniu wykonania poprzedniego rozkazu
- 106. IT1A\_W08,IT1A\_U08** W procesorze wykorzystującym superskalarność  
Przykładowo odp.) możliwe jest jednoczesne wykonanie więcej niż jednej instrukcji
- 107. IT1A\_W08,IT1A\_U08** Sumator jednobitowy  
Przykładowo odp.) jest układem sekwencyjnym
- 108. IT1A\_W08,IT1A\_U08** Rejestr rozkazów  
Przykładowo odp.) przechowuje kod rozkazu wczytany z pamięci programu
- 109. IT1A\_W08,IT1A\_U08** Przykłady układów kombinacyjnych to  
Przykładowo odp.) multiplekser oraz transkoder
- 110. IT1A\_W08,IT1A\_U08** Przykłady układów sekwencyjnych to  
Przykładowo odp.) żaden z powyższych
- 111. IT1A\_W08,IT1A\_U08** Transmisja asynchroniczna  
Przykładowo odp.) układy nadawczy i odbiorczy synchronizowane są wspólną dodatkową linią z sygnałem synchronicznym

## 7 Metody numeryczne

**112. EKK\_1,EKK\_7** W pewnym hipotetycznym binarnym systemie zmiennoprzecinkowym zakres danych ujemnych wynosi  $\langle -b, -a \rangle$ , chcemy zapisać liczbę  $c$ , która jest liczbą mniejszą od  $-b$  i która ma nieskończone rozwinięcie. W związku tym zastępujemy ją najbliższą liczbą, którą da się zapisać w tym systemie, czyli liczbą  $-b$ . Z jakim błędem numerycznym mamy tutaj do czynienia:

Przykładowo odp.) Błędem obcięcia

**113. EKK\_1,EKK\_7** Warunkiem koniecznym i wystarczającym zbieżności metod iteracyjnych prostych (takich jak takich jak metoda Jacobiego czy metoda Gaussa-Seidla) rozwiązywania układów równań liniowych:

Przykładowo odp.) Promień spektralny macierzy iterowanej w danej metodzie jest zawsze mniejszy od 1.

**114. EKK\_1,EKK\_7** Do metod nazywanych metodami dokładnymi rozwiązywania układów równań liniowych zalicza się:

Przykładowo odp.) Metoda Jacobiego

**115. EKK\_1,EKK\_7** Które z poniżej wymienionych zagadnień numerycznych wykorzystują właściwości przybliżania funkcji wielomianem interpolującym:

Przykładowo odp.) Rozwiązywanie układów równań metodami iteracyjnymi

**116. EKK\_1,EKK\_7** Macierz Hilberta osiąga wysokie wartości współczynnika uwarunkowania (ang. Condition number) na tej podstawie możemy stwierdzić, że:

Przykładowo odp.) Macierz Hilberta jest dobrze uwarunkowana

**117. EKK\_1,EKK\_7** Wielomiany sklejące (ang. spline) trzeciego stopnia muszą spełniać następujące warunki w punktach sklejeń:

Przykładowo odp.) Przechodzenie funkcji interpolującej przez węzły interpolacji

**118. EKK\_1,EKK\_7** Należy wskazać zdania prawdziwe dotyczące zagadnienia interpolacji wielomianowej z wykorzystaniem jednomianów (tzw bazy naturalnej):

Przykładowo odp.) Jest to zadanie źle uwarunkowane

**119. EKK\_1,EKK\_7** Błędy związane z ograniczeniem nieskończonego ciągu wymaganych obliczeń do skończonej liczby działań nazywamy:

Przykładowo odp.) Błędami obcięcia (ang. truncation errors)

**120. EKK\_1,EKK\_7** Jeśli niewielkie względne zaburzenia danych wejściowych powodują niewielkie względne zmiany wyników to wówczas

Przykładowo odp.) Współczynnik uwarunkowania osiąga wysoką wartość

**121. EKK\_1,EKK\_7** Warunkami wystarczającymi, gwarantującymi zbieżność poszukiwania miejsc zerowych funkcji  $f(x)$  metodą bisekcji są:

Przykładowo odp.) Na końcach przedziału  $[a, b]$  wartości funkcji  $f(x)$  przyjmują przeciwne znaki, czyli zachodzi  $f(a)f(b) < 0$

**122. EKK\_1,EKK\_7** Stosując algorytm stycznych poszukiwania jednokrotnego miejsca zerowego funkcji  $f(x)$  w przedziale domkniętym  $[a, b]$  w dostatecznej bliskości pierwiastka uzyskujemy zbieżność:

Przykładowo odp.) Wykładniczą

**123. EKK\_1,EKK\_7** Do całkowania numerycznego używa się m.in. kwadratur Newtona – Cotesa. Do prostych kwadratur Newtona – Cotesa należą:

Przykładowo odp.) Metoda trapezów

**124. EKK\_1,EKK\_7** Efekt Rungego jest charakterystyczny dla następujących metod interpolacji:

Przykładowo odp.) Interpolacji Newtona dla węzłów równoodległych

**125. EKK\_1,EKK\_7** Które zdania dotyczące Metody Eliminacji Gaussa rozwiązywania układów równań są prawdziwe:

Przykładowo odp.) Nie wymaga przekształcenia układu równań do postaci z macierzą trójkątną górną

**126. EKK\_1,EKK\_7** Aby wyeliminować lub znacząco ograniczyć efekt Rungego przy zadaniu interpolacji można:

Przykładowo odp.) Zastosować interpolację funkcjami sklejanymi zamiast metody Lagrange'a

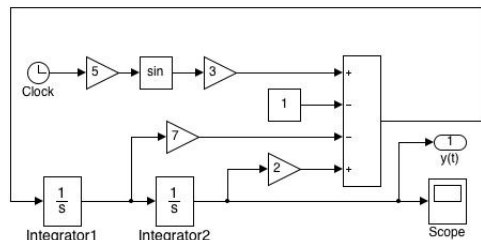
## 8 Analiza numeryczna i symulacja systemów

**127. EKK\_1** Wskaż prawidłowo sformułowane warunki w zagadnieniach początkowych Cauchy'ego (IVP) dla równania różniczkowego  $y'(t) = f(t, y(t))$ ,  $f: \Omega \subset \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ ,  $t \in [a, b]$ .

Przykładowo odp.)  $n = 2, y_{10} = y_1(a), y_{20} = y_2(b)$ .

**128. EKK\_1, EKK\_18** Wskaż diagramy SIMULINKa, które reprezentują równanie różniczkowe  $y'' - 2y' + 7y = 3 \sin(5t) - 1$

Przykładowo odp.)



**129. EKK\_1** Które zdania odnoszące się do metod rozwiązywania zagadnień początkowych dla równań różniczkowych są prawdziwe?

Przykładowo odp.) Metody Adamsa-Moultona są metodami niejawnymi.

**130. EKK\_1** Numeryczne rozwiązywanie zagadnienia początkowego. Która metoda jest metodą samostarującą: Przykładowo odp.) każda jawna metoda Rungego-Kutty.

**131. EKK\_1** W przypadku metody Eulera zastosowanej do rozwiązywania zagadnienia początkowego dla  $y'(t) = f(t, y(t))$ ,  $y_0 = y(0)$  (przy założeniu braku błędu numerycznego wszystkich operacji arytmetycznych) Przykładowo odp.) Może się zdarzyć, że błąd rozwiązania jest równy zero, ale nigdy nie ma takiej gwarancji,

**132. EKK\_1** Numeryczne rozwiązywanie zagadnienia początkowego. W metodach typu predyktor-korektor (PECE)

Przykładowo odp.) stosuje się albo metodę jawną albo niejawną.

**133. EKK\_1** Które zdania, odnoszące się do metod Rungego-Kutty (RK) rozwiązywania zagadnienia początkowego dla równań różniczkowych, są prawdziwe:

Przykładowo odp.) Algorytm 4. etapowej jawnej metody RK jest identyczny jak 4. etapowej niejawnej metody RK.

**134. EKK\_1** Jawne metody Rungego-Kutty (RK). Niech  $n_e$  oznacza liczbę etapów metody, a  $r$  - maksymalny osiągalny rząd metody.

Przykładowo odp.)  $n_e \geq r$ ,

**135. EKK\_1** Algorytmy optymalizacji statycznej.

Przykładowo odp.) Metoda Newtona wymaga obliczania w każdym kroku gradientu i hesjanu.

**136. EKK\_1, EKK\_U19** Dyskretna aproksymacja średniokwadratowa.

Dla  $n + 1$  wartości zmiennej niezależnej  $x_i, i = 0, 1, \dots, n$   $x_i < x_{i+1}$  wykonano pomiary i otrzymano  $n + 1$  wartości  $y_i$ . Zależność wielkości mierzonej od  $x$  aproksymowano wielomianem  $W_m(x) = \sum_{i=0}^m a_{i,m} x^i$  z błędem aproksymacji  $E_m$ . Proszę zaznaczyć prawdziwe implikacje

Przykładowo odp.)  $m < n \Rightarrow E_m > 0$

**137. EKK\_1, EKK\_U19** Dla  $n + 1$  wartości zmiennej niezależnej  $x_i, i = 0, 1, \dots, n$   $x_i < x_{i+1}$  wykonano pomiary i otrzymano  $n + 1$  wartości  $y_i$ . Zależność wielkości mierzonej od  $x$  aproksymowano wielomianem  $W_m(x) = \sum_{i=0}^m a_{i,m} x^i$ . Rozważamy 3 sposoby obliczania błędu aproksymacji  $E_m$ :

- $E_m = \min_{a_0, \dots, a_m} \sum_{i=0}^n |y_i - W_m(x_i)|$



$$2. E_m = \min_{a_0, \dots, a_m} \sum_{i=0}^n (y_i - W_m(x_i))^2$$

$$3. E_m = \min_{a_0, \dots, a_m} \max_{i=0, \dots, n} |y_i - W_m(x_i)|$$

Obliczenie współczynników  $a_i$  można sprowadzić do zagadnienia liniowego  
Przykładowo odp.) w żadnym spośród 1-3.

**138. EKK\_1** Dla tych samych danych eksperymentalnych

$i$	0	1	2
$x_i$	2	4	6
$y_i$	1	2	1

wyznaczono 3 funkcje aproksymujące. W każdym przypadku  $k = 1, 2, 3$  funkcja aproksymująca miała postać  $f_k(x) = a_k x + b_k$ , ale użyto innego kryterium jakości aproksymacji :

$$1. \text{ Dla } k = 1 : \min_{a_1, b_1} \sum_{i=0}^2 |y_i - f_1(x_i)|$$

$$2. \text{ Dla } k = 2 : \min_{a_2, b_2} \sum_{i=0}^2 (y_i - f_2(x_i))^2$$

$$3. \text{ Dla } k = 3 : \min_{a_3, b_3} \max_{i=0, 1, 2} |y_i - f_3(x_i)|.$$

Proszę zaznaczyć prawidłowe odpowiedzi:

Przykładowo odp.)  $a_1 = a_2 = a_3, b_1 = \frac{3}{2}, b_2 = \frac{3}{2}, b_3 = \frac{4}{3}$ .

**139. EKK\_1** Numeryczne metody optymalizacji.

Rozważmy funkcję kwadratową  $n$  zmiennych, (w zapisie wektorowym  $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$ )

$$f(\mathbf{x}) = \mathbf{x}^T A \mathbf{x} + \mathbf{b}^T \mathbf{x} + c$$

gdzie  $A$  jest macierzą  $n \times n$ , a  $\mathbf{b}$  wektorem  $n \times 1$  o stałych współczynnikach.  $c$  jest skalar. Załóżmy, że macierz  $A$  jest dodatnio określona. Funkcja  $f$  ma minimum w punkcie  $\mathbf{x}_{min}$ .

Aby znaleźć minimum tej funkcji mamy do dyspozycji 3 metody: simpleksu Neldera-Meada, najszybszego spadku (*steepest descent*) oraz Newtona. Startujemy z dowolnego punktu  $\mathbf{x} \in \mathbb{R}^n, \mathbf{x} \neq \mathbf{x}_{min}$ .

Przykładowo odp.) Wszystkie 3 wymienione metody gwarantują znalezienie minimum funkcji  $f$  w pierwszym kroku.

**140. EKK\_1** Dyskretna aproksymacja średniokwadratowa.

Czy obliczanie parametrów (współczynników) funkcji aproksymującej można sprowadzić do rozwiązania układu równań liniowych?

Przykładowo odp.) Nie.

**141. EKK\_1** Aproksymacja dyskretna.

Do aproksymacji zbioru punktów  $P_i(x_i, y_i), i = 0, 1, \dots, n$  używamy funkcji  $f(x)$  o parametrach  $a_j, j = 0, 1, \dots, m$ . Stosując 3 różne kryteria jakości aproksymacji (miary błędu aproksymacji)

$$1. k = 1 : \min_{a_0, \dots, a_m} \sum_{i=0}^n |y_i - f^{(1)}(x_i)|$$

$$2. k = 2 : \min_{a_0, \dots, a_m} \sum_{i=0}^n (y_i - f^{(2)}(x_i))^2$$

$$3. k = 3 : \min_{a_0, \dots, a_m} \max_{i=0, 1, \dots, n} |y_i - f^{(3)}(x_i)|.$$

otrzymujemy trzy funkcje aproksymujące  $f^{(k)}(x), k = 1, 2, 3$  różniące się między sobą tylko wartościami parametrów  $a_j, j = 0, 1, \dots, m$ .

Niech  $\Delta_{max}^{(k)}$  oznacza odległość (w sensie metryki maksimum)  $k$ -tej funkcji aproksymującej  $f^{(k)}$  od punktu najbardziej oddalonego, tzn.  $\Delta_{max}^{(k)} = \max_{i=0, \dots, n} |y_i - f^{(k)}(x_i)|$ . Proszę zaznaczyć prawdziwe relacje

Przykładowo odp.)  $\Delta_{max}^{(2)} \geq \Delta_{max}^{(3)}$ ,

## 9 Języki i technologie webowe

**142. IT1A\_W10,IT1A\_U10** Zaznacz prawdziwe stwierdzenia. Droga pakietu w sieci Internet pomiędzy dwoma węzłami, tj. lista adresów węzłów odwiedzanych przez pakiet:

Przykładowo odp.) może być nieskończona

**143. IT1A\_W10,IT1A\_U10** Serwery DNS oferują:

Przykładowo odp.) translację nazw symbolicznych do ich adresów IP

**144. IT1A\_W10,IT1A\_U10** Zaznacz prawdziwe stwierdzenie. Protokół HTTP w wersji 1.1

Przykładowo odp.) umożliwia transmisję dowolnych typów danych

**145. IT1A\_W10,IT1A\_U10** Do bezpośredniej komunikacji z serwerem WWW służą następujące narzędzia:

Przykładowo odp.) host

**146. IT1A\_U05,IT1A\_U03** Wskaż prawdziwe stwierdzenia o poniższym fragmencie kodu XHTML 1.0 Strict.

```
1 <p><a href=http://www.agh.edu.pl><br></p>
```

Przykładowo odp.) Nie jest poprawny, wartość atrybutu *href* musi być umieszczona w cudzysłowie.

**147. IT1A\_U05,IT1A\_U03** Dany jest poniższy fragment kodu XHTML 1.0 Strict.

```
1 
```

Obrazek *i.jpg* ma rozmiary 1024x768. Zaznacz prawdziwe stwierdzenia.

Przykładowo odp.) Atrybuty *width* i *height* są niedopuszczalne w standardzie XHTML 1.0 Strict.

**148. IT1A\_U05,IT1A\_U03** Ile zasobów z dyrektywami CSS może być skojarzonych z pojedynczym dokumentem XHTML 1.0 Strict?

Przykładowo odp.) Więcej niż jeden.

**149. IT1A\_U05,IT1A\_U03** Zaznacz prawdziwe stwierdzenia dotyczące poniższego kodu CSS 2.1.

```
1 .nav > div {
2     color: white;
3     background: #119500;
4     float: right;
5     width: 120px;
6     padding: 1px;
7     font-size: small;
8     border: solid red 1px;
9 }
```

Przykładowo odp.) Element jest opływany; umieszczony z prawej strony.

**150. IT1A\_W10,IT1A\_W06, IT1A\_U05, IT1A\_U03, IT1A\_U10** Wskaż prawdziwe stwierdzenia odnośnie poniższego fragmentu kodu PHP.

```
1 $fp = fopen("plik_do_blokowania", "r+");
2 if (flock($fp, LOCK_EX)) {
3     processing();
4     flock($fp, LOCK_UN);
5 } else {
6     problem();
7 }
8 fclose($fp);
```

Przykładowo odp.) Linia 3 nie będzie wykonywana współbieżnie.

**151. IT1A\_W10,IT1A\_W06, IT1A\_U05, IT1A\_U03, IT1A\_U10** Zawartość poniższego formularza przesłano do skryptu PHP. Zaznacz prawdziwe stwierdzenia.

```
1 <form action="skrypt.php" method="post"
2     enctype="multipart/form-data">
3     <p>
4         <input type="file" name="plik" />
5         <input type="text" name="comment" />
6         <input type="submit" value="wyslij" />
7     </p>
8 </form>
```

Przykładowo odp.) W zmiennej `$_FILES['plik']` znajduje się zawartość przesłanego pliku.

**152. IT1A\_W10,IT1A\_W06, IT1A\_U05, IT1A\_U03, IT1A\_U10** Co jest efektem działania poniższego programu w języku PHP.

```
1 <?php
2 $wiek=array('ala' => 12,'ela' => 22,'franek' => 54);
3 foreach ($wiek as $k => $w )
4     echo $k.' '. $w." \n";
5 ?>
```

Przykładowo odp.) Wygenerowanie na standardowym wyjściu pojedynczej linii; linia jest ciągiem bajtów zakończonych znakiem końca linii.

**153. IT1A\_W10,IT1A\_W06, IT1A\_U05, IT1A\_U03, IT1A\_U10** Jak długi będzie czas wykonania poniższego programu napisanego w języku PHP? Zakłada się, że program uruchamiany jest jako aplikacja WWW tj. dostępny jest pod określonym adresem URI, a interpreter PHP uruchamiany jest przez serwer WWW.

```
1 <?php
2     echo 'start';
3     sleep(6);
4 ?>
```

Przykładowo odp.) Co najmniej 6 sekund, ale może być krótszy, w zależności od skonfigurowanego maksymalnego dozwolonego czasu wykonywania aplikacji.

**154. IT1A\_W10,IT1A\_W06, IT1A\_U05, IT1A\_U03** Która z poniższych metod w języku JavaScript zwraca element o unikalnym identyfikatorze *form*?

Przykładowo odp.) `document.getElementById('form')`

**155. IT1A\_W10,IT1A\_W06, IT1A\_U05, IT1A\_U03** Jaki jest efekt uruchomienia poniższego kodu w języku JavaScript zakładając, że został on umieszczony między elementami w dokumencie XHTML?

```
1 car=new Array();
2 car[0]=new Object();
3 car[0].make='Fiat';
4 car[0].vin='123';
5 car[1]=new Object();
6 car[1].make='Ford';
7 car[1].vin='456';
8
9 for ( idx in car ) {
10     for ( prop in car[idx] ) {
11         document.write(car[idx][prop]);
12     }
13 }
```

13 } }

Przykładowo odp.) W miejscu umieszczenia skryptu w dokumencie XHTML zostanie wygenerowany ciąg bajtów: *makevinmakevin*.

**156. IT1A\_W10, IT1A\_W06, IT1A\_U05, IT1A\_U03** Zaznacz prawdziwe stwierdzenia dotyczące poniższego kodu w języku JavaScript.

```
1 function updateAjax () {
2   xmlhttp = new XMLHttpRequest();
3   xmlhttp.onreadystatechange = function () {
4     if (xmlhttp.readyState==4 && xmlhttp.status==200) {
5       document.getElementById("stime").innerHTML=xmlhttp.responseText;
6     }
7   }
8   xmlhttp.open("GET","date.php",true);
9   xmlhttp.send();
10  window.setTimeout("updateAjax()",1000);
11 }
12 window.setTimeout("updateTime(); updateAjax();",5000);
```

Przykładowo odp.) Po wywołaniu funkcji *updateAjax()* zostanie ona wykonana ponownie po upływie 1 sekundy.

**157. IT1A\_W05, IT1A\_U05** Dany jest dokument XML oraz odpowiednie DTD. Zaznacz prawdziwe stwierdzenia.

Przykładowo odp.) Aby sprawdzić poprawność strukturalną (ang. *validity*) konieczne jest DTD.

## 10 Badania operacyjne i teoria złożoności obliczeniowej

158. M\_W001 Która z poniższych złożoności obliczeniowych jest wykładnicza:

Przykładowo odp.)  $\mathcal{O}(n^{100})$

159. M\_U003 Które z poniższych zdań jest fałszywe.

Przykładowo odp.) Złożoność czasowa algorytmu *pre-order* przeszukiwania drzewa jest wielomianowa.

160. M\_U003 Co przyjmujemy zazwyczaj jako górne ograniczenie w algorytmach podziału i ograniczeń?

Przykładowo odp.) Ilość wygenerowanych podproblemów

161. M\_U002 W algorytmach ewolucyjnych stosowane są różne rodzaje reprodukcji. Która z nich polega na wybieraniu najlepszych osobników z wylosowanych podzbiorów?

Przykładowo odp.) Reprodukacja proporcjonalna

162. M\_U003 Do znalezienia minimalnego czasu wykonania przedsięwzięcia reprezentowanego poprzez graf (sieć) stosuje się metodę ścieżki krytycznej. Na czym polega ta metoda?

Przykładowo odp.) Na wyznaczeniu najdłuższej ścieżki prowadzącej z wierzchołka początkowego do wierzchołka końcowego

163. M\_U002 Dla której z podstawowych technik obliczeń ewolucyjnych charakterystyczna jest adaptacja zasięgu mutacji?

Przykładowo odp.) Dla algorytmów genetycznych

164. M\_W002 Dany jest pierwotny program liniowy postaci:

$$\mathbf{c}^T \mathbf{x} \rightarrow \max, \mathbf{A} \cdot \mathbf{x} \leq \mathbf{b}, \mathbf{x} \geq 0.$$

Program dualny do niego ma postać:

Przykładowo odp.)  $\mathbf{b}^T \mathbf{y} \rightarrow \min, \mathbf{A}^T \cdot \mathbf{y} \geq \mathbf{c}, \mathbf{y} \geq 0.$

165. M\_U003 Co nazywamy mostem grafu?

Przykładowo odp.) Krawędź, której usunięcie spowoduje wzrost liczby składowych spójności grafu

166. M\_U003 Jak nazywamy podzbiór  $V' \subset V$  zbioru wierzchołków grafu  $G = (V, E)$ , taki, że każdy węzeł nienależący do  $V'$  jest sąsiedni do pewnego elementu z  $V'$ ?

Przykładowo odp.) Zbiór dominujący

167. M\_W004 Jak nazywamy system obsługi zadań, w którym każde zadanie musi przejść przez wszystkie maszyny w jednakowym, ściśle określonym porządku?

Przykładowo odp.) System gniazdowy

168. M\_U001 W algorytmie symulowanego wyżarzania z sąsiedztwa bieżącego rozwiązania bazowego losuje się jedno rozwiązanie. Co się dzieje, jeżeli jest ono gorsze od dotychczasowego rozwiązania bazowego?

Przykładowo odp.) Zawsze zostaje odrzucone i nie staje się nowym rozwiązaniem bazowym

169. M\_W001 W teorii złożoności wszystkie problemy decyzyjne, które w wielomianowym czasie rozwiązuje niedeterministyczna maszyna Turinga, tworzą pewną klasę problemów. Jak brzmi jej nazwa?

Przykładowo odp.) Klasa problemów pseudowielomianowych

170. M\_W002 Zastosowanie metody programu dualnego pozwala na:

Przykładowo odp.) Przejście od modelu opisanego układem równań nieliniowych do modelu liniowego.

171. M\_W001 Dane są algorytmy A i B o złożonościach odpowiednio  $\mathcal{O}_A(n^3)$  i  $\mathcal{O}_B((\log n)^3)$ . Oba algorytmy wywołano dla pewnych danych wejściowych:  $a$  (dla A) i  $b$  (dla B). Szybciej (w sensie czasu mierzonego w sekundach) wykona się algorytm:

Przykładowo odp.) B

**172. MU002** W jakim celu w algorytmach ewolucyjnych stosuje się funkcję kary?  
Przykładowo odp.) Wyskalowania funkcji przystosowania

## 11 Sieci komputerowe

**173. M\_W001,M\_W002,M\_W003** Adres typu broadcast (rozgłoszenia) IP w wersji 4 dla sieci IP, w której znajduje się host 110.104.1.10 i którą określa maska 255.0.0.0, to:  
Przykładowo odp.) 110.104.1.255

**174. M\_W001,M\_W002,M\_W003** Pole o nazwie Time to live w datagramie IP, które zabezpiecza przed zapełnieniem rutowania datagramu pomiędzy kolejnymi ruterami w sieci, zawiera:  
Przykładowo odp.) Czas w sekundach, w którym datagram IP można jeszcze przekazywać dalej.

**175. M\_W001,M\_W002,M\_W003** Nazwa ramki stosowanej w technologii IEEE 802.11 i emitowanej przez urządzenie Access Point i stosowanej między innymi w celu propagowania informacji o sieci bezprzewodowej, to:  
Przykładowo odp.) Beacon

**176. M\_W001,M\_W002,M\_W003** Protokół UDP definiuje identyfikatory przesyłanych do hosta-odbiorcy datagramów zwane numerami portów, o długości:  
Przykładowo odp.) 32 bity

**177. M\_W001,M\_W002,M\_W003** Wartości adresu IPv6 oraz maski, określające wszystkie hosty w Internecie, to:  
Przykładowo odp.) 255::255/128

**178. M\_W001,M\_W002,M\_W003** Istnienie zasady “Longest prefix match“ w rutowaniu IP spowoduje, że adres docelowy 200.200.200.1 datagramu IP przy istnieniu w tablicy rutowania jednocześnie reguł o wzorcach i maskach (podano w notacji CIDR): 200.200.200.0/18, 200.200.200.0/20, 200.200.200.0/22, 200.200.200.0/24 zostanie dopasowany do:  
Przykładowo odp.) 200.200.200.0/18

**179. M\_W001,M\_W002,M\_W003** Maksymalna długość pakietu IP wersja 4, licząc w bajtach, to:  
Przykładowo odp.) 1024

**180. M\_W001,M\_W002,M\_W003** Określenie stosowane wobec rutera MPLS (MultiProtocol Label Switching), będącego w danej sytuacji odbiorcą datagramów z etykietami MPLS od innego (nie będącego przedmiotem rozważań), to:  
Przykładowo odp.) Designated router

**181. M\_W001,M\_W002,M\_W003** Ruter iBGP (internal Border Gateway Protocol), którego wprowadzenie do systemu rutowania iBGP umożliwia znaczne zredukowanie ilości otwartych sesji BGP pomiędzy innymi ruterami (rezygnację z tzw. Full-mesh) nosi nazwę:  
Przykładowo odp.) BGP Mirror

**182. M\_W001,M\_W002,M\_W003** Liczba klas CoS (Class of Service), definiowanych przez podstawowy mechanizm implementacji QoS (Quality of Service) w Ethernet (czyli standard IEEE 802.1p), to:  
Przykładowo odp.) 65536

**183. M\_W001,M\_W002,M\_W003** Wariant protokołu STP (Spanning Tree Protocol, IEEE 802.1d) pozwalający w technologii Ethernet na logiczne grupowanie sieci VLAN (Virtual LAN) i budowanie mniejszej liczby drzew rozpinających (po jednym Spanning Tree dla każdej zdefiniowanej grupy), to:  
Przykładowo odp.) PVSTP (Per VLAN Spanning Tree Protocol)

**184. M\_W001,M\_W002,M\_W003** Rodzaje (grupy) urządzeń fizycznych definiowanych w technologii ZigBee, to:  
Przykładowo odp.) ZigBee End Device, ZigBee Coordinator, ZigBee Router

**185. M\_W001,M\_W002,M\_W003** Nazwa procesu przekazywania wiedzy o trasach pomiędzy różnymi protokołami rutowania dynamicznego IP w ruterach IP, to:

Przykładowo odp.) Redystrybucja

**186. M\_W001,M\_W002,M\_W003** Symbole literowe, określające rodzaje popularnych w sieciach komputerowych wtyków światłowodowych, to:

Przykładowo odp.) DC, SS, S2

**187. M\_W001,M\_W002,M\_W003** Co określa standard IEEE 802.1Q?

Przykładowo odp.) Private VLAN nadbudowaną nad Ethernet

**188. M\_W001,M\_W002,M\_W003** Protokół umożliwiający konwersję adresu IP zdalnej stacji na jej adres MAC w Ethernet, to:

Przykładowo odp.) MLD (Multicast Listener Discovery)

**189. M\_W001,M\_W002,M\_W003** Co zawiera pole Extended Unique Identifier (EUI) w adresie IPv6?

Przykładowo odp.) Losową, choć unikatową w skali całego segmentu sieci IPv6 liczbę

**190. M\_W001,M\_W002,M\_W003** Domyślna wartość metryki Administrative Distance w tablicy rutowania IP ruterów (np. Cisco, Juniper, Hewlett Packard) przewidziana dla protokołu RIP (Routing Information Protocol), to:

Przykładowo odp.) 15

**191. M\_W001,M\_W002,M\_W003** W technologii Fibre Channel (stotowanej w sieciach SAN) port przełącznika Switch Fabric mogący pracować w topologii pętli arbitrażowej (pętli z arbitrażem) sieci Fibre Channel, to port typu:

Przykładowo odp.) N

**192. M\_W001,M\_W002,M\_W003** Dwie pod-warstwy definiowane w ramach warstwy drugiej modelu ISO-OSI to odpowiednio:

Przykładowo odp.) LP (Link Pulse) i PHY (Physical)

**193. M\_W001,M\_W002,M\_W003** Zadana w jednostce dBm efektywną moc wypromieniowaną (Effective Isotropic Radiated Power, EIRP) bezprzewodowego urządzenia nadawczego stosowanego w technologii sieciowej na podstawie mocy wypromieniowanej P zadanej w watach można obliczyć stosując wzór:

Przykładowo odp.)  $EIRP = 1 / (P * 1mW)$

**194. M\_W001,M\_W002,M\_W003** Jednostka wysokości urządzenia sieciowego montowanego w standardzie RACK wynosząca 1,75 cala (44,45 mm) oznaczana jest symbolem:

Przykładowo odp.) RACK

**195. M\_W001,M\_W002,M\_W003** Rodzaj obszaru (area) w domenie OSPF (Open Shortest Path First) nie otrzymującego żadnych informacji o zewnętrznych (external) trasach rutowania OSPF, to:

Przykładowo odp.) NSSA

**196. M\_W001,M\_W002,M\_W003** Parametr o nazwie "Wielkość okna"(Window size), którego wartość przekazywana jest w datagramach potwierdzenia TCP (Transmission Control Protocol Acknowledgment) w kierunku odbiorcy do nadawcy ma na celu:

Przykładowo odp.) Określenie ilości danych w datagramie, w którym się znajduje - i w przypadku potwierdzeń TCP nie ma żadnego znaczenia

**197. M\_W001,M\_W002,M\_W003** Dwa rodzaje obszarów (area) w protokole rutowania dynamicznego IS-IS (Intermediate System to Intermediate System), to:

Przykładowo odp.) Autonomous System i Internal System



## 12 Paradygmaty programowania

**198. IT1A\_W05,EKK\_3** Podstawowym, deklaratywnym językiem programowania logicznego jest: Przykładowo odp.) PROLOG

**199. IT1A\_W05** Które z poniższych mechanizmów są wbudowane w interpreterze języka PROLOG: Przykładowo odp.) Szukanie z nawrotami

**200. IT1A\_W05** Rozważmy następującą definicję predykatu `member/2`:

```
1 member(H, [H|_]).
2 member(H, [_|_]):- member(H, _).
```

Dla wywołania `member(X,[0,1,[2,3],4])` interpreter zwróci:  
Przykładowo odp.) 4 rozwiązania

**201. IT1A\_W05** Rozważmy następującą definicję predykatu `member/2`:

```
1 member(H, [H|_]).
2 member(H, [_|_]):- member(H, _).
```

Dla wywołania `member(X, [0, 1, 2, 1, 3, 1, 4])` interpreter zwróci:  
Przykładowo odp.) 7 rozwiązań, w tym 3 identyczne

**202. IT1A\_W05** Rozważmy następującą definicję predykatu `append/3` do łączenia list:

```
1 append([], L, L).
2 append([H|_], L, [H|_]) :- append(_, L, _).
```

Dla wywołania `append(L1, L2, [1, 2, 3, 4, 5])` . interpreter zwróci:  
Przykładowo odp.) 4 różne odpowiedzi

**203. IT1A\_W05** Rozważmy następującą definicję predykatu `append/3` do łączenia list:

```
1 append([], L, L).
2 append([H|_], L, [H|_]) :- append(_, L, _).
```

Aby pobrać ostatni element zadanej listy L (np. L = [1,2,3,4] i powiązać go ze zmienną E (np. E=4) należy zastosować zapytanie:

Przykładowo odp.) `append(_, [E], L)` .

**204. IT1A\_W05** Rozważmy następujący program w PROLOGU:

```
1 p(a).
2 p(b).
3 p(c).
4 p(a).
5 p(c).
6
7 run :-
8     p(X),
9     assert(q(X)),
10    fail.
```

Po skompilowaniu i wykonaniu programu z wywołaniem `run`:

Przykładowo odp.) W pamięci zapisane zostanie 5 faktów: `q(a)` . `q(b)` . `q(c)` . `q(a)` . `q(c)` .

**205. IT1A\_W05** Rozważmy następujący program w PROLOGU:

```
1 ln (0, []) :- !.
2 ln (N, [N|L]) :- N1 is N-1, ln (N1, L).
```

Po skompilowaniu i wykonaniu programu z wywołaniem `ln (7, L)`:  
Przykładowo odp.) Dostaniemy wynik `L = [7, 6, 5, 4, 3, 2, 1]`

**206. IT1A\_W05** Rozważmy następujący program w PROLOGU:

```
1 s1 (X):- not (p(X)), !, q(X).
2 s2 (X):- q(X), not (p(X)).
3 p(a).
4 q(b).
```

Po skompilowaniu i wykonaniu programu:  
Przykładowo odp.) `s1(X)` zwraca `a`; `s2(X)` zwraca `b`

**207. IT1A\_W05** Rozważmy następujące propozycje programów iteracyjnego sumowania elementów zadanej listy w PROLOGU: Poprawny jest program:  
Przykładowo odp.)

```
1 loop ([], S, S).
2 loop ([H|T], A, S) :-
3     AH is A+H,
4     loop (T, AH, S).
```

**208. IT1A\_W05** Jaki typ w Haskellu będzie miało następujące wyrażenie: `r x = x:r x`  
Przykładowo odp.) `r :: Eq a => [a]`

**209. IT1A\_W05** Jak wygląda poprawna wartość dla typu `data Tree a = L a | N (Tree a) a (Tree a)`  
Przykładowo odp.) `N (L 5) 5 (L 2)`

**210. IT1A\_W05** Haskell jest językiem opartym o paradygmat  
Przykładowo odp.) zdarzeniowy

**211. IT1A\_W05** Mechanizm typów w języku Haskell jest  
Przykładowo odp.) dynamiczny

**212. IT1A\_W05** Zaznacz prawdziwe zdania.  
Przykładowo odp.) Można używać funkcyjnego stylu programowania w języku JavaScript.

**213. IT1A\_W05** Funkcje wyższego rzędu w programowaniu funkcyjnym to  
Przykładowo odp.) potoczne określenie funkcji trudnych w implementacji

**214. IT1A\_W05** Jaki mechanizm w językach funkcyjnych pozwala na wykonanie operacji na zbiorze danych?  
Przykładowo odp.) pętla

**215. IT1A\_W05** Zaznacz prawdziwe zdania dotyczące programowania funkcyjnego.  
Przykładowo odp.) Funkcyjny styl programowania można uprawiać w ograniczonym zakresie w językach imperatywnych jak C albo JavaScript.

## 13 Programowanie mikrokontrolerów i mikroprocesorów

- 216. IT1A\_W17** Ile rejestrów 8-bitowych dostępnych dla programisty znajduje się w procesorach z rodziny x86?  
Przykładowo odp.) 8
- 217. IT1A\_W17, IT1A\_U08** Jaki tryb adresowania wykorzystuje rozkaz ADDL (%ebx),%eax?  
Przykładowo odp.) natychmiastowy
- 218. IT1A\_W17, IT1A\_U08** Jaka instrukcja jest równoważna w działaniu do instrukcji SHL \$1,%eax?  
Przykładowo odp.) ROL \$1,%eax
- 219. IT1A\_W17, IT1A\_U08** Która z poniższych instrukcji dotyczy operacji na blokach danych?  
Przykładowo odp.) CMC
- 220. IT1A\_W17** Według jakiej reguły może być dokonywana konwersja do liczby całkowitej w jednostce FPU (Floating Point Unit)?  
Przykładowo odp.) round to nearest
- 221. IT1A\_W17, IT1A\_U08** Ile razy wykona się pętla zbudowana w oparciu o instrukcję LOOP, jeśli przed jej rozpoczęciem zawartość rejestru %ecx była równa 0?  
Przykładowo odp.)  $2^{32}$
- 222. IT1A\_W17, IT1A\_U08** Ile razy zawartość rejestru %ah zostanie zapisana do pamięci poprzez użycie instrukcji REP STOSB, jeżeli przed jej wykonaniem zawartość rejestru %ecx była równa  $x$ ?  
Przykładowo odp.)  $x$
- 223. IT1A\_W17, IT1A\_U08** Jaka będzie zawartość rejestru %eax po sekwencji rozkazów?
- ```
1 MOVL $0xFFFF0000,%eax
2 NEG %eax
```
- Przykładowo odp.) 0x00000000
- 224. IT1A\_W17, IT1A\_U08** Jaka będzie zawartość rejestru %al po sekwencji rozkazów?
- ```
1 MOVW $0xFF00,%ax
2 ADCB %ah,%al
3 ADCB %ah,%al
```
- Przykładowo odp.) nieokreślona
- 225. IT1A\_W17, IT1A\_W19, IT1A\_U21, IT1A\_U24** Na jakim rodzaju schematu pokazane są połączenia elektryczne w układzie opartym na mikrokontrolerze?  
Przykładowo odp.) organizacyjnym
- 226. IT1A\_W17, IT1A\_W19** W jakim rodzaju pamięci mikrokontrolera użytkownik zwykle zapisuje kod programu?  
Przykładowo odp.) DRAM
- 227. IT1A\_W17, IT1A\_W19** Jakie elementy występujące w mikrokontrolerach nie występują w mikroprocesorach?  
Przykładowo odp.) RTC
- 228. IT1A\_W03** Czy język maszynowy jest tożsamy z językiem asemblera?  
Przykładowo odp.) tak
- 229. IT1A\_W03** Jakie narzędzie służy do zamiany kodu napisanego w języku asemblera na kod maszynowy?  
Przykładowo odp.) linker

**230. IT1A\_U12, IT1A\_U17, IT1A\_U21** Które z narzędzi nie umożliwia stworzenia kodu na mikrokontroler z rodziny AVR?

Przykładowo odp.) Atmel Studio

## 14 Systemy operacyjne

- 231. EKK\_IT1A\_W06** Która wypowiedź odnosi się do pamięci asocjacyjnej:  
Przykładowo odp.) Adres danych jest dostarczany przez użytkowników
- 232. EKK\_IT1A\_W09,EKK\_IT1A\_W06** Dla uniknięcia błędów uwarunkowanych czasowo, maksymalna liczba procesów które mogą znajdować się wewnątrz sekcji krytycznej wynosi  
Przykładowo odp.) 1
- 233. EKK\_IT1A\_W06** Strategia, która pozwala procesowi, który spełnia warunki wykonalności być chwilowo zawieszonym jest nazywana:  
Przykładowo odp.) strategią „first come first served”
- 234. EKK\_IT1A\_W06** Stan uprzywilejowany jest:  
Przykładowo odp.) jest stanem procesu, który otrzyma procesor
- 235. EKK\_IT1A\_W09,EKK\_IT1A\_W06** Komunikacja między procesami  
Przykładowo odp.) jest wymagana dla wszystkich procesów
- 236. EKK\_IT1A\_W06** Przy organizacji pamięci wirtualnej dynamiczna translacja adresu  
Przykładowo odp.) wymaga przechowywania stron w specyficznej lokalizacji na dysku
- 237. EKK\_IT1A\_W09,EKK\_IT1A\_W06** Inicjalna wartość semafora uogólnionego implementującego sekcję krytyczną wynosi:  
Przykładowo odp.) true
- 238. EKK\_IT1A\_W09,EKK\_IT1A\_W06** Proces transferowania danych, które mają być docelowo wyprowadzone na urządzenie peryferyjne do przestrzeni pamięci pomocniczej i transferowanie ich na to urządzenie w dogodniejszym czasie nosi nazwę:  
Przykładowo odp.) spooling
- 239. EKK\_IT1A\_W09,EKK\_IT1A\_W06** Problem producent-konsument może być rozwiązany przy pomocy  
Przykładowo odp.) warunkowych rejonów krytycznych
- 240. EKK\_IT1A\_W06** Centralny Procesor, po otrzymaniu informacji o przerwaniu z urządzenia wejścia/wyjścia  
Przykładowo odp.) przekazuje kontrolę do magistrali adresowej i szyny danych do urządzenia przerywającego
- 241. EKK\_IT1A\_W09,EKK\_IT1A\_W06** Który z problemów rozwiązuje zaproponowany przez Dijkstrę algorytm Bankiera  
Przykładowo odp.) wykluczania zakleszczenia (deadlock exclusion)
- 242. EKK\_IT1A\_W06** Jeżeli wirtualny adres w programie jest 16 bitowy i rozmiar strony jest 0,5 K to możemy maksymalnie zaadresować następującą liczbę stron:  
Przykładowo odp.) 32
- 243. EKK\_IT1A\_W09,EKK\_IT1A\_W06** System operacyjny jest:  
Przykładowo odp.) zbiorem driverów obsługujących urządzenie wejścia wyjścia (input-output devices)
- 244. EKK\_IT1A\_W09,EKK\_IT1A\_W06** W systemie zarządzania pamięcią rejestry graniczne DATUM i LIMIT  
Przykładowo odp.) wyznaczają początek i koniec programu
- 245. EKK\_IT1A\_W09,EKK\_IT1A\_W06** Jeżeli system operacyjny chce wykonywać więcej niż jeden program w danym momencie czasu to musi :  
Przykładowo odp.) zapewnić przetwarzanie równoległe

**246. EKK\_IT1A\_W06** Szyfrowanie kluczem publicznym w szyfrowaniu asymetrycznym  
Przykładowo odp.) pozwala jedynie właściwemu nadawcy odkodować komunikat

**247. EKK\_IT1A\_W06** Buforowanie plików realizowane jest w celu:  
Przykładowo odp.) przyśpieszenie dostępu do pamięci głównej

## 15 Inżynieria oprogramowania

**248. EKK\_1,EKK\_2** Spośród poniższych wskaż prawidłowe przepływy danych występujące w diagramie DFD.  
Przykładowo odp.) Magazyn do procesu

**249. EKK\_1,EKK\_2** Jakimi cechami charakteryzuje się dobre oprogramowanie  
Przykładowo odp.) niezawodność, wydajność, poprawność

**250. EKK\_1,EKK\_2** Podczas którego etapu procesu wytwarzania oprogramowania przygotowywane są definicje bazy danych oraz plików?  
Przykładowo odp.) Analizy

**251. EKK\_1,EKK\_2** Wstępna próba zdefiniowania elementów systemu oraz ich wzajemnych relacje, organizowanie tych elementów w dobrze określone warstwy z wyraźnych nakreślonymi zależnościami nazywa się analizą  
Przykładowo odp.) przypadków użycia

**252. EKK\_1,EKK\_2** Celem testowania oprogramowania jest  
Przykładowo odp.) zdefiniowanie jakości kodu

**253. EKK\_1,EKK\_2** Wskaż które stwierdzenia są prawidłowe  
Przykładowo odp.) przepływ danych w DFD może być dwukierunkowy

**254. EKK\_1,EKK\_2** Tworzenie modelu obiektowego z istniejącej relacyjnej bazy danych jest określane jako  
Przykładowo odp.) Backward engineering

**255. EKK\_1,EKK\_2** Model wymagania składa się z czterech części  
Przykładowo odp.) Przypadki użycia, opis interfejsu, diagram klas, zakres projektu

**256. EKK\_1,EKK\_2** Którego z poniższych narzędzi nie używa się podczas analizy systemowej?  
Przykładowo odp.) Decision Tree

**257. EKK\_1,EKK\_2** Stosowanie techniki prototypowania jest odpowiednie dla  
Przykładowo odp.) aplikacji w których duży nacisk położono na interfejs użytkownika

**258. EKK\_1,EKK\_2** Które z poniższych nie jest przedmiotem zainteresowania w fazie projektowania?  
Przykładowo odp.) Zakres projektu

**259. EKK\_1,EKK\_2** Jakie są główne aktywności w modelu spiralnym?  
Przykładowo odp.) Planowanie, Analiza Ryzyka, Konstrukcja, Walidacja

**260. EKK\_1,EKK\_2** Czego nie znajdziemy w diagramie kontekstowym?  
Przykładowo odp.) Procesu

**261. EKK\_1,EKK\_2** Najważniejszy celem inżynierii oprogramowania jest tworzenie software które jest  
Przykładowo odp.) Wszystkie powyższe

**262. EKK\_1,EKK\_2** Przykładem ryzyka występującego podczas wytwarzania oprogramowania jest  
Przykładowo odp.) wszystkie powyższe odpowiedzi są poprawne

**263. EKK\_1,EKK\_2** Które z poniższych wielkości są mierzalne bezpośrednio?

1. rozmiar produktu
2. wysiłek potrzebny do wytworzenia software
3. harmonogram
4. jakość

Przykładowo odp.) zarówno (1) i (2)



## 16 Programowanie współbieżne i rozproszone

**264. IT1A\_W05,IT1A\_W06** Jak wygląda poprawna definicja obiektu funkcyjnego w języku Erlang?

Przykładowo odp.)  $F1(X) \rightarrow X+1$ .

**265. IT1A\_W05,IT1A\_W06** Jaki będzie wynik operacji w Erlangu:  $[1,2,3] \text{ -- } [3,2,3,5]$ .

Przykładowo odp.)  $[1]$

**266. IT1A\_W05,IT1A\_W06** System typów w Erlangu jest:

Przykładowo odp.) dynamiczny - sprawdzany w trakcie wykonania

**267. IT1A\_W05,IT1A\_W06** W jaki sposób tworzy się proces w języku Erlang wykonujący funkcję F1?

Przykładowo odp.)  $\text{Pid} \rightarrow \text{exec}(F1)$ .

**268. IT1A\_W05,IT1A\_W06** Jak w języku Erlang przesyła się wiadomość (Mesg) do procesu posiadając jego identyfikator (Pid)?

Przykładowo odp.)  $\text{Pid send Mesg}$ .

**269. IT1A\_W06** Jaki model jest użyty do komunikacji między procesami w języku Erlang?

Przykładowo odp.) Model hybrydowy.

**270. IT1A\_W05,IT1A\_W06** Jak zrealizowana jest komunikacja między procesami w języku Erlang?

Przykładowo odp.) Jest oparta na kolejkach FIFO.

**271. IT1A\_W05,IT1A\_W06** Jaki będzie wynik wykonania następującej instrukcji w języku Erlang:

$\text{lists :map}(\text{fun}(X) \rightarrow \{X,X+1\} \text{ end}, [1,2,3])$  ?

Przykładowo odp.)  $\{\{1,2\},\{2,3\},\{3,4\}\}$

**272. IT1A\_W05,IT1A\_W06** Jaka jest funkcja obiektu chronionego w Adzie?

Przykładowo odp.) Przyspieszenie działania programu.

**273. IT1A\_W05,IT1A\_W06** Jakie operacje są możliwe do zdefiniowania dla typu kontrolowanego w Adzie?

Przykładowo odp.) Tylko operator przypisania.

**274. IT1A\_W05,IT1A\_W06** W jaki sposób określa się kierunek przekazywania parametrów z/do procedur w języku Ada?

Przykładowo odp.) Słowa kluczowe inbound i outbound w deklaracji parametrów.

**275. IT1A\_W05,IT1A\_W06** Jaki jest rodzaj typizacji w języku Ada?

Przykładowo odp.) dynamiczny

**276. IT1A\_W05,IT1A\_W06** Jak komunikują się zadania w języku Ada?

Przykładowo odp.) Przez przesyłanie wiadomości.

**277. IT1A\_W05,IT1A\_W06** Które z wymienionych algorytmów służą do wyboru lidera w systemie rozproszonym?

Przykładowo odp.) Algorytm tyrana.

**278. IT1A\_W06** Zaznacz prawdziwe zdania.

Przykładowo odp.) Prawo Amdahla pozwala oszacować teoretyczny wzrost szybkości systemu przy zmianie liczby procesorów.

## 17 Bazy danych

**279. IT1A\_W11, IT1A\_U11** Wskaż wszystkie prawdziwe stwierdzenia dotyczące kluczy w relacyjnym modelu danych.

Przykładowo odp.) Dla każdej zmiennej relacyjnej można wyznaczyć klucz.

**280. IT1A\_W11, IT1A\_U11** Dana jest relacja  $R$  o schemacie  $H = \{A, B, C, D, E, F, G\}$  i zbiorze zależności funkcyjnych  $F = \{\{C\} \rightarrow \{A\}, \{C\} \rightarrow \{B, F\}, \{C\} \rightarrow \{G\}, \{E\} \rightarrow \{C\}, \{G\} \rightarrow \{A, B\}\}$ . Które z podanych zbiorów są kluczami relacji  $R$ ?

Przykładowo odp.)  $\{A, B, C, D, E, F, G\}$

**281. IT1A\_W11, IT1A\_U11** Dla których z podanych operacji algebry relacji schemat(y) argumentu (ów) i relacji wynikowej są takie same?

Przykładowo odp.) selekcja

**282. IT1A\_W11, IT1A\_U11** Załóżmy, że w zapytaniu opartym na dwóch relacjach zastępujemy operator złączenia wewnętrznego operatorem złączenia zewnętrznego. Wskaż te operatory, których użycie gwarantuje wynik nie mniejszy (w sensie relacji inkluzji) niż użycie operatora złączenia wewnętrznego.

Przykładowo odp.) złączenie lewostronne zewnętrzne

**283. IT1A\_W11, IT1A\_U11** Wskaż, w których przypadkach klauzule instrukcji *select* są ułożone we właściwej kolejności.

Przykładowo odp.) from, group by, having, where

**284. IT1A\_W11, IT1A\_U11** Wskaż, które elementy dopuszczalne w konceptualnym modelu danych są niekompatybilne z modelem relacyjnym.

Przykładowo odp.) związki rekurencyjne wiele do wielu

**285. IT1A\_W11, IT1A\_U11** Dana jest relacja  $R$  o schemacie  $H = \{A, B, C, D, E, F\}$  i zbiorze zależności funkcyjnych  $F = \{\{A\} \rightarrow \{B\}, \{C\} \rightarrow \{D, E\}, \{A, C\} \rightarrow \{F\}\}$ . Które z podanych dekompozycji relacji  $R$  na relacje o schematach  $H_1$ ,  $H_2$  i  $H_3$  są bezstratne?

Przykładowo odp.)  $H_1 = \{A, B\}$ ,  $H_2 = \{C, D\}$ ,  $H_3 = \{E, F\}$

**286. IT1A\_W11, IT1A\_U11** Wskaż wszystkie prawdziwe stwierdzenia dotyczące postaci normalnej Boyce'a–Codda.

Przykładowo odp.) Dowolna relacja dwuatrybutowa jest w BCNF.

**287. IT1A\_W11, IT1A\_U11** Dana jest relacja  $R$  o schemacie  $H = \{A, B, C, D, E\}$  oraz zbiór zależności funkcyjnych  $F = \{\{B, C\} \rightarrow \{D, E\}, \{C, D\} \rightarrow \{B, E\}, \{D\} \rightarrow \{C\}, \{E\} \rightarrow \{B\}\}$ . W jakiej maksymalnie postaci normalnej jest relacja  $R$ ? (Zakładamy, że jest w 1NF.)

Przykładowo odp.) BCNF

**288. IT1A\_W11, IT1A\_U11** Wskaż wszystkie prawdziwe stwierdzenia dotyczące trzeciej postaci normalnej.

Przykładowo odp.) Jeżeli relacja jest w 3NF, to jest również w BCNF.

**289. IT1A\_W11, IT1A\_U11** Wskaż wszystkie prawdziwe stwierdzenia dotyczące kluczy obcych w relacyjnym modelu danych.

Przykładowo odp.) Wartości klucza obcego mogą się powtarzać.

**290. IT1A\_W11, IT1A\_U11** Wskaż wszystkie prawdziwe stwierdzenia dotyczące użycia funkcji agregujących w systemie PostgreSQL.

Przykładowo odp.) W klauzuli *group by* można użyć atrybuty, które nie występują z klauzuli *select*.

**291. IT1A\_W11, IT1A\_U11** Wskaż wszystkie prawdziwe stwierdzenia dotyczące transakcji.

Przykładowo odp.) Transakcja jest ciągiem operacji w bazie danych, które należy wykonać wszystkie lub nie wykonywać żadnej z nich.

**292. IT1A\_W11, IT1A\_U11** Wskaż, które ograniczenia można definiować na poziomie kolumny (w instrukcji *create table*).

Przykładowo odp.) proste klucze główne

**293. IT1A\_W11, IT1A\_U11** Wskaż wszystkie prawdziwe stwierdzenia dotyczące wartości *null*.

Przykładowo odp.) W modelu relacyjnym wartość *null* jest traktowana jako trzecia różna od *false* i *true* wartość logiczna.

## 18 Lingwistyka formalna i automaty

**294. K\_W15, K\_W03, K\_W04, K\_W07, K\_U15, K\_U21, K\_U22, K\_K01, K\_K06** Gramatyka jest wieloznaczna, jeżeli

Przykładowo odp.) jest to gramatyka kontekstowa lub bezkontekstowa

**295. K\_W15, K\_W03, K\_W04, K\_W07, K\_U15, K\_U21, K\_U22, K\_U03, K\_U04** Które z poniższych napisów są prawdziwe dla języka generowanego przez następującą gramatykę  $G = \langle \{Q, R, X\}, \{\Delta, \nabla\}, \{X \rightarrow \nabla \Delta R, X \rightarrow \Delta \nabla Q, R \rightarrow \Delta \nabla X, R \rightarrow \Delta \nabla, Q \rightarrow \nabla \Delta X, Q \rightarrow \nabla \Delta\}, X \rangle$ :

Przykładowo odp.)  $\nabla \Delta \Delta \nabla \nabla \Delta \nabla \Delta \Delta \nabla$

**296. K\_W15** Dla domknięcia Kleene'ego prawdziwe są następujące stwierdzenia:

Przykładowo odp.) jest to zbiór wszystkich słów otrzymany poprzez złożenie dowolnej długości słów

**297. K\_W15** Zapis  $L^* = \bigcup_{i=0}^{\infty} L^i$  oznacza dla języków:

Przykładowo odp.) operację złożenia w nieskończoności

**298. K\_W15, K\_W03, K\_W04, K\_W07, K\_K01, K\_K06** Dla klasyfikacji gramatyk Chomsky'ego prawdziwe są następujące stwierdzenia:

Przykładowo odp.) praktyczne znaczenie dla możliwości konstruowania kompilatorów języków programowania mają gramatyki klasy 2 i 3

**299. K\_W15, K\_W03, K\_W04, K\_W07, K\_K01, K\_K06** Dla języków i gramatyk formalnych, odnośnie postaci normalnej Chomsky'ego oraz postaci normalnej Greibach można sformułować następujące stwierdzenia (duże litery alfabetu łacińskiego to symbole nieterminalne, a litery małe to symbole terminalne):

Przykładowo odp.) dla każdego języka bezkontekstowego istnieje gramatyka w postaci normalnej Chomsky'ego

**300. K\_W15, K\_U15, K\_U21, K\_U22** Odnośnie lematu o pompowaniu dla języków regularnych prawdziwe są następujące stwierdzenia:

Przykładowo odp.) lemat służy pokazaniu, że określone języki nie są regularne

**301. K\_W15, K\_K01, K\_K06** Jeżeli  $Lin$  oznacza gramatyki liniowe,  $BK$  gramatyki bezkontekstowe,  $Reg$  gramatyki regularne,  $PL$  gramatyki prawostronnie liniowe, a  $LL$  gramatyki lewostronnie liniowe, to które z następujących relacji są prawdziwe:

Przykładowo odp.)  $Lin \subseteq BK$

**302. K\_W15, K\_W03, K\_W04, K\_W07, K\_U15, K\_U21, K\_U22** Które ogólne stwierdzenia odnośnie języków, gramatyk i automatów są prawdziwe:

Przykładowo odp.) jeżeli  $L$  jest językiem bezkontekstowym, to istnieje automat ze stosem akceptujący ten język i jest on mu równoważny

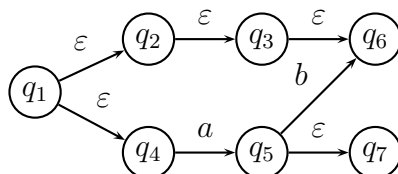
**303. K\_W15, K\_W03, K\_W04, K\_W07, K\_U15, K\_U21, K\_U22** Dla danego ustalonego języka  $L$  i alfabetu  $V$ , językiem ilorazowym  $L/x$  nazywamy język postaci:

$$L/x = \{y \in V^* : xy \in L\}$$

dla  $x \in V^*$ . Które stwierdzenia są prawdziwe:

Przykładowo odp.) dla dowolnych  $x, y \in V^*$  mamy  $L/xy = (L/x)/y$

**304. K\_W15, K\_W03, K\_W04, K\_W07, K\_U15, K\_U21, K\_U22, K\_U03, K\_U04**  $\varepsilon$ -domknięciem  $E$  dla stanu początkowego  $q_1$  dla przedstawionego poniżej automatu



są zbiory

Przykładowo odp.)  $E(q_1) = \{q_1, q_4, q_5, q_6\}$

**305. K\_W15, K\_W03, K\_W04, K\_W07, K\_U15, K\_U21, K\_U22** Dany jest automat niedeterministyczny  $A = \{S = \{A, B, C\}, V = \{0, 1\}, \{\delta(A, 1) = B, \delta(A, 1) = C, \delta(B, 0) = A, \delta(C, 0) = B\}, s_0 = A, Z = \{C\}\}$

Automat po determinizacji (w znaczeniu algorytmu Rabina-Scotta) będzie miał:

Przykładowo odp.) osiem stanów

**306. K\_W15, K\_W03, K\_W04, K\_W07, K\_U15, K\_U21, K\_U22** Jeżeli  $r$  oraz  $s$  są wyrażeniami regularnymi dla języków odpowiednio  $R$  oraz  $S$ , to  $(r+s)$ ,  $rs$  i  $r^*$  są wyrażeniami regularnymi reprezentującymi odpowiednio zbiory:

Przykładowo odp.)  $R \cup S$ ,  $RS$  i  $R^*$

**307. K\_W15, K\_W03, K\_W04, K\_W07, K\_U15, K\_U21, K\_U22** Wyrażenie regularne  $(0 + 1)^*00(0 + 1)^*$  opisuje:

Przykładowo odp.) łańcuchy, w których tylko raz wystąpiło podwojenie zer

**308. K\_W15, K\_W03, K\_W04, K\_W07, K\_U15, K\_U21, K\_U22** Mamy języki  $L_1 = \{a^{2^n} : n > 0\}$  oraz  $L_2 = \{a^{2n} : n > 0\}$ . Które z tych języków są regularne?

Przykładowo odp.)  $L_1$  – nie,  $L_2$  – nie

## 19 Teoria kompilacji i kompilatory

- 309. K\_W15, K\_W03, K\_W04, K\_W07, K\_W20, K\_K01, K\_K06** Typowy skaner języka formalnego ma za zadanie  
Przykładowo odp.) usunąć komentarze
- 310. K\_W15, K\_W03, K\_W04, K\_W07, K\_W20, K\_K01, K\_K06** Typowy parser języka formalnego ma za zadanie  
Przykładowo odp.) usunąć komentarze zagnieżdżone w innych komentarzach
- 311. K\_W15, K\_W03, K\_W04, K\_W07, K\_W20, K\_K01, K\_K06** Przez rozbiór kanoniczny rozumiemy rozbiór, który  
Przykładowo odp.) w pierwszej kolejności redukuje prawostronne symbole formy zdaniowej
- 312. K\_W15, K\_W03, K\_W04, K\_W07, K\_W20, K\_U24** Metoda generacyjna rozbioru gramatycznego polega na tym, że  
Przykładowo odp.) wprowadzany najpierw brakujące symbole nieterminalne formy zdaniowej
- 313. K\_W15, K\_W03, K\_W04, K\_W07, K\_W20, K\_U24** Metoda redukcyjna rozbioru gramatycznego polega na tym, że  
Przykładowo odp.) redukuje się liczbę możliwych nawrotów poprzez zastosowanie gramatyki jednoznacznej
- 314. K\_W15, K\_W03, K\_W04, K\_W07, K\_W20, K\_U24** Dla analizatorów klasy  $LL(k)$  prawdziwe są następujące stwierdzenia:  
Przykładowo odp.) parametr  $k$  oznacza liczbę błędów, po których analizator przerywa działanie
- 315. K\_W15, K\_W03, K\_W04, K\_W07, K\_W20, K\_U24** Dla analizatorów klasy  $LL(k)$  prawdziwe są następujące stwierdzenia:  
Przykładowo odp.) nie wykonują nawrotów
- 316. K\_W15, K\_W03, K\_W04, K\_W07, K\_W20, K\_U24** Usunięcie  $\varepsilon$ -produkcji z gramatyki klasy  $G_{LL(k)}$  powoduje  
Przykładowo odp.) nie jest zasadniczo możliwe dla gramatyki jednoznacznej i bezkontekstowej
- 317. K\_W15, K\_W03, K\_W04, K\_W07, K\_W20, K\_U24** W odniesieniu do parserów klasy  $LR(k)$  prawdziwe są następujące ogólne stwierdzenia:  
Przykładowo odp.) jeśli parsing jest kontynuowany, to może nastąpić zamiana wierzchu stosu na prawą stronę produkcji
- 318. K\_W15, K\_W03, K\_W04, K\_W07, K\_W20, K\_U24** W odniesieniu do pracy parserów klasy  $LR(k)$  i funkcji *action* prawdziwe są stwierdzenia:  
Przykładowo odp.) funkcja *action* przyjmuje wartości ze zbioru  $\{shift, reduce, accept, error\}$
- 319. K\_W15, K\_W03, K\_W04, K\_W07, K\_W20, K\_U24** Dla tablic sterujących parserów klasy  $LR(0)$  i przykładowej produkcji  $A \rightarrow XYZ$  mamy:  
Przykładowo odp.) trzy możliwe sytuacje
- 320. K\_W15, K\_W03, K\_W04, K\_W07, K\_W20, K\_U24** Budowa tablic sterujących dla analizatorów klasy  $LR$  może stwarzać pewne trudności, szczególnie w zakresie automatyzacji, co ma pośredni wpływ na istnienie wielu odmian tych parserów. Które z poniższych prostych stwierdzeń są poprawne:  
Przykładowo odp.) pierwsza litera w nazwie  $SLR$  oznacza *Sift*
- 321. K\_W15, K\_W03, K\_W04, K\_W07, K\_W20, K\_U24** Dla pewnej gramatyki mówimy, że sytuacja  $LR(0)$  oznaczona  $[N \rightarrow \beta_1.\beta_2]$  dla  $\gamma \in V^*$  jest poprawna, gdy przy założeniu  $\alpha\beta_1 = \gamma$  prawdziwe jest:  
Przykładowo odp.)  $S \xRightarrow{rm*} \alpha N \omega \xRightarrow{rm} \alpha \beta_1 \beta_2 \omega$

**322. K\_W15, K\_W03, K\_W04, K\_W07, K\_W20, K\_U24** Pomędzy parserami  $LR$  zachodzą następujące relacje w odniesieniu do zbiorów gramatyk:

Przykładowo odp.)  $LR(0) \subset SLR(1) \subset LALR(1) \subset LR(1)$

**323. K\_W15, K\_W03, K\_W04, K\_W07, K\_W20, K\_U24** Porównując gramatyki  $LL$  oraz  $LR$  można powiedzieć, że:

Przykładowo odp.) gramatyki  $LR$  opisują szerszą klasę niż  $LL$

## 20 Podstawy sztucznej inteligencji

**324. EKK\_1,EKK\_3** Który (które) z poniższych algorytmów zapewniają znalezienie najkótszej ścieżki w grafie (koszt każdego łuku równy 1):

Przykładowo odp.) Algorytm przeszukiwania włąb (ang. Depth-First Search)

**325. EKK\_1,EKK\_2** Algorytm Tree-Search Breadth-First F wygenerował 400 węzłów do głąbokości 3. Szacunkowy (zastępczy) *branching factor*  $b$  wynosi:

Przykładowo odp.) Około 7

**326. EKK\_1,EKK\_2** Aby algorytm  $A^*$  znajdował rozwiązanie optymalne w literaturze przytaczane są następujące wymagania co do funkcji heurystycznej  $h(n)$ :

Przykładowo odp.) funkcja  $h(n)$  powinna być monotoniczna (ang. monotonic)

**327. EKK\_1,EKK\_2** Algorytmy Genetyczne (AG) stosowane są do optymalizacji złożonych funkcjonatów, w tym problemów z ograniczeniami; które własności tych algorytmów są prawdziwe: Dla wywołania `member(X, [0, 1, 2, 1, 3, 1, 4])` interpreter zwróci:

Przykładowo odp.) AG gwarantują znalezienie rozwiązania dopuszczalnego

**328. EKK\_1,EKK\_2** Dla problemu kryptoarytmetycznego SEND+MORE=MONEY najlepsze zgrubne ale optymistyczne oszacowanie ilości rozwiązań do przebadania to:

Przykładowo odp.) Mniej niż 1 000 000 rozwiązań potencjalnych

**329. EKK\_1,EKK\_2** Rozważmy wieże hanojskie o  $N$  kręgach. Przestrzeń stanów i rozwiązanie optymalne mają:

Przykładowo odp.)  $2^N$  stanów oraz  $3^N$  kroków

**330. EKK\_1,EKK\_2** Rozważmy zadanie programowania z ograniczeniami, gdzie szukane są wartości trzech zmiennych  $X$ ,  $Y$ , oraz  $Z$ . Wszystkie zmienne są różne od siebie (`alldifferent([X, Y, Z])`). Dziedzina  $X$  to  $\{1, 2, 3\}$ , dziedzina  $Y$  to  $\{2, 3, 4\}$  a dziedzina  $Z$  to  $\{3, 4, 5\}$ . Ile istnieje rozwiązań dopuszczalnych:

Przykładowo odp.) 13

**331. EKK\_1,EKK\_2** Logiczną kosekwencją zbioru zdań:

$$\{\neg A \vee P, \neg P \vee B \vee D, \neg D \vee N, \neg D \vee M, \neg D \vee H, \neg H \vee \neg S \vee R, \neg H \vee R \vee I, A, \neg B, \neg R\}$$

nie jest:

Przykładowo odp.)  $N$