

# Wstęp do informatyki

## Grupa 1

1. Warunek jednoznaczności w kontekście algorytmów określa, iż

- (a) w każdym kroku mamy skończoną ilość alternatywnych ścieżek do wyboru;
- (b) nie ma wątpliwości co do kolejnego kroku jaki zostanie wykonany; <- tak
- (c) algorytm jednoznacznie stosuje się tylko do pewnej klasy zadań.

2. Do typów prostych zaliczamy

- (a) typ liczbowy, typ znakowy i typ logiczny; <- tak
- (b) typ liczbowy, typ znakowy i rekord;
- (c) typ liczbowy, tablicę i rekord.

3. Zaznacz (jako pseudokod na którym pokazano nieprawidłowe wykorzystanie tablicy dwuwymiarowej `tab`)

```
\<-- tak, nie, tak
a)          b)          c)
tab[a] := 13.5  tab[1][x] := tab[x][1]  tab[1][] := tab[][1]
```

4. Dla poniższego typu rekordowego

```
type obiekt is record
  Integer: dlugosc
  Real: masa
end
```

zaznacz poprawny pseudokod

```
<-- tak, tak, nie
a)          b)          c)
tab is array[10] of obiekt  obiekt: obj1  obiekt: o
...          ...          ...
tab[0].masa := 134          x := obj1.dlugosc  o := masa[134]
```

5. Jeśli do słownika wstawione zostaną elementy `a`, `b` oraz `c` wówczas pobierać będziemy z niego elementy w kolejności

- (a) `a`, `b`, `c`
- (b) `c`, `b`, `a`
- (c) nie można określić kolejności pobierania – słownik nie określa kolejności dla przechowywanych elementów. <- tak

6. Instrukcja warunkowa służy do

- (a) powtarzania fragmentu kodu;
- (b) wykonania alternatywnego fragmentu kodu; <- tak
- (c) wykonania jednego z dwóch wzajemnie wykluczających się fragmentów kodu. <- tak

7. Zaznacz poprawne wywołanie funkcji z dwoma argumentami

- (a) `x := function xyz(a, b)`
- (b) `x := xyz(5, b) <- tak`
- (c) `x := xyz[5][b]`

8. Łączna liczba wywołań funkcji dla poniższej funkcji `foo`

```
function foo(n)
  if (n = 0 or n = 1) then
    return 1

  return foo(n-1)*foo(n-2)
```

przy wywołaniu `foo(4)` wynosi

- (a) 5

- (b) 7
- (c) 9 <- tak

9. W wyniku wykonania poniższego pseudokodu

```
function trelemorele(x)
begin
  if x > 3 then
  begin
    return 3
  end
  else
  begin
    return 1
  end
  return 2
end
```

funkcja `trelemorele` wywołana z argumentem 5 zwróci

- (a) 1;
- (b) 2;
- (c) 3. <- tak

10. Zaznacz poprawne wyrażenia

- (a)  $\overline{(x \cdot y)} = \bar{x} + \bar{y}$  <- tak
- (b)  $\overline{(x \cdot y)} = \bar{x} \cdot \bar{y}$
- (c)  $\overline{(x \cdot y)} = \overline{(x + y)}$

11. Poniższe wyrażenie będzie prawdą dla parzystych wierszy dwuwymiarowej tablicy `tab`

- (a) `tab[i] mod 2 = 0`
- (b) `tab[i][j] mod 2 = 0`
- (c) `tab[i][] mod 2 = 0`

12. Zaznacz poprawne słowa języka  $L$  nad alfabetem  $\Sigma = \{a, b, c, d, e\}$  określonego przez poniższe reguły

**R1**  $c$  należy do  $L$ .

**R2** Każdy niepusty ciąg znaków który nie zawiera  $a$  oraz  $b$  i nie zaczyna się od  $c$  należy do  $L$ .

**R3** Ciąg znaków zawierający  $a$  należy do  $L$  wtedy i tylko wtedy gdy występuje tylko jedno  $a$  i rozdziela ono poprawne słowa języka  $L$ .

**R4** Ciąg znaków zawierający  $b$ , ale nie zawierający  $a$  jest słowem języka  $L$  jeśli  $b$  rozdziela poprawne słowa języka  $L$ .

- (a) `cdebedcacdd`
- (b) `debdaded` <- tak
- (c) `bdeadeebe`

13. Biorąc pod uwagę poniższy zbiór reguł BNF

```
<b> := <b><d>
<b> := <d>
<d> := a
<d> := c
```

można powiedzieć, że

- (a)  $a$  jest symbolem terminalnym, <- tak
- (b)  $b$  jest symbolem terminalnym,
- (c) nie ma symboli terminalnych.

14. BNF to skrót od:

- (a) Best Notation Format
- (b) Backus-Naur Form <- tak
- (c) Best to kNow Format

15. BNF jest

- (a) powszechnie używana do zapisu składni języków programowania i protokołów komunikacyjnych. <- tak
- (b) powszechnie używana do zapisu operacji Boolowskich.
- (c) powszechnie używana do gramatycznego upraszczania wyrażeń logicznych.

16. Element występujący co najmniej jeden raz, który przy wykorzystaniu EBNF może być zapisany jako  $\langle a \rangle := \{\langle b \rangle\}^+$  będzie zapisany w BNF jako

- (a)  $\langle b \rangle := \langle a \rangle$   
 $\langle a \rangle := \langle a \rangle \langle b \rangle$
- (b) <- tak  
 $\langle a \rangle := \langle b \rangle$   
 $\langle a \rangle := \langle a \rangle \langle b \rangle$
- (c)  $\langle b \rangle := \langle a \rangle$   
 $\langle b \rangle := \langle a \rangle \langle b \rangle$

17. Wyrażenie regularne  $a(b|cd?)e$  pasuje do

- (a)  $abe$  <- tak
- (b)  $acd?e$
- (c)  $acde$  <- tak

18. Maszynę Turinga określa

- (a) zbiór stanów, alfabet, funkcja przejścia, wyróżniony stan początkowy; <- tak
- (b) dostępna długość taśmy, liczba określająca ilość stanów oraz wektor przejścia;
- (c) funkcja przejścia określająca ile symboli alfabetu może zostać zapisanych w jednej komórce na taśmie.

19. Maszyna Turinga zdefiniowana przez poniższą funkcję przejścia

	S		K
---+-----+-----			
a	S;b;R		
b	S;a;R		Stop
c	K;c;L		

i rozpoczynająca działanie ze stanu S dla taśmy  $cba[a]bbaac$  (element nad którym znajduje się głowica, w tym przypadku a, znajduje się w nawiasach kwadratowych [])

- (a) nigdy się nie zatrzyma;
- (b) zatrzyma się po odczytaniu symbolu c; <- tak
- (c) zatrzyma się po wykonaniu 9 ruchów.

20. W pozycyjnym systemie liczbowym o podstawie 2

- (a) nie da się zapisać liczb większych niż 256;
- (b) można zapisać tylko liczby całkowite;
- (c) liczby ułamkowe zapisywane są przy pomocy kodowanie U2.

21. Zaznacz prawdziwe zdanie

- (a) 1 bit to 8 bajtów.
- (b) Bit może przyjąć 256 różnych wartości.
- (c) Na 5 bitach można zapisać 16 różnych wartości. <- i tak i nie, zależy jak interpretować

22. Binarna liczba zapisana na 3 bitach z których jedynie najmłodszy ma wartość równą 1 ma wartość

- (a) 1 <- tak
- (b) 4

(c) 7

23. 1KiB jest

- (a) większe niż 1kB; <- tak
- (b) równe 1kB;
- (c) mniejsze niż 1kB.

24. 111 Kib równe jest

- (a) 14208B <- tak
- (b) 13KiB + 896B <- tak
- (c) 14kB + 208B <- tak

25. Strona kodowa

- (a) określa sposób przypisania poszczególnym liczbom znaków (zarówno drukowalnych jak i znaków sterujących). <- tak
- (b) służy do konwersji pomiędzy ASCII a UTF8.
- (c) pozwala zamieniać liczby całkowite na ich kod U2.

26. BOM w UTF służy do

- (a) (ang. *Big Order Modifier*) określa sposób modyfikowania znaków za pomocą sekwencji znaków sterujących;
- (b) (ang. *Best Order Markup*) określa preferowany kierunek pisma w kodowaniu Unicode;
- (c) (ang. *Byte Order Mark*) określenia kolejności bajtów w UTF. <- tak

27. Przy poniżej określonym kodowaniu

3 pierwsze bity - określają ilość bitów (n) używanych na zapis liczb

n następujących bitów - określają wysokość obrazka

i dalej

n - określają szerokość obrazka

2 bity - określają rodzaj obiektu:

0 okrąg - kodowany jest na za pomocą trzech kolejnych liczb kodujących  
odpowiednio współrzędne x i y środka oraz promień

1 odcinek - kodowany jest na za pomocą czterech kolejnych liczb kodujących  
odpowiednio współrzędne x i y początku i końca odcinka

2 wybór koloru - kolorowi białemu przypisana jest wartość 0, czarnemu 1,  
czerwonemu 2

3 koniec

ciąg szesnastkowo zapisanych bajtów: 4F F2 11 56 AC 30 przedstawia

- (a) biały odcinek (5,6) – (10,3);
- (b) czerwony okrąg  $s = (5,6), r = 3$ ;
- (c) czarny odcinek (5,6) – (10,12).

28. Jeśli zmienna  $a$  przyjmuje wartość 13 a zmienna  $b$  wartość -5 to po wykonaniu poniższej sekwencji działań

$a = a \text{ xor } b$

$b = b \text{ xor } a$

$a = a \text{ xor } b$

prawdziwe będzie zdanie

- (a) zmienna  $a$  ma wartość 13 a zmienna  $b$  wartość 5.
- (b) zmienna  $a$  ma wartość -13 a zmienna  $b$  wartość -5.
- (c) zmienna  $a$  ma wartość -5 a zmienna  $b$  wartość 13. <- tak

29. Operacje bitowe: zaznacz prawdziwe zdanie

- (a)  $153 \text{ and } 150 = 144$  <- tak
- (b)  $153 \text{ or } 150 = 144$

(c)  $153 \text{ and } 150 = 159$

30. Zaznacz zdanie prawdziwe

- (a) liczby ujemne w notacji znak-moduł i uzupełnienie do dwóch mają taką samą postać.
- (b) liczby ujemne w notacji znak-moduł i uzupełnienie do dwóch mają taką samą postać z wyjątkiem bitu znaku, który jest przeciwny.
- (c) liczby dodatnie w notacji znak-moduł i uzupełnienie do dwóch mają taką samą postać. <- tak

31. Zaznacz zdanie prawdziwe

- (a) W arytmetyce zmiennoprzecinkowej dodawanie nie jest łączne, tzn  $(x + y) + z \neq x + (y + z)$  <- tak
- (b) Zero nie jest wartością specjalną i może zostać bezpośrednio reprezentowana w formacie zmiennoprzecinkowym.
- (c) Na zadanej liczbie  $n$  bitów można zapisać więcej liczb rzeczywistych niż całkowitych.

32. Jeśli do stosu dodam elementy pobrane z kolejki a następnie z tablicy `tab` wypiszę elementy o indeksach wskazanych przez liczby pobierane ze stosu to wówczas przy poniższych założeniach

```
do kolejki dodaję elementy: 2, 4, 1, 3
tab[1] = x
tab[2] = y
tab[3] = z
tab[4] = w
```

wypisane zostanie

- (a) `z x w y` <- tak
- (b) `z w x y`
- (c) `z x y w`

33. Zaznacz zdanie prawdziwe

- (a) Kolejka jest strukturą w której do wskazywania elementów używamy indeksów liczbowych.
- (b) W kolejce jest określona kolejność, ale nie można odwołać się do dowolnego elementu. <- tak
- (c) Kolejka jest strukturą w której do wskazywania elementów używamy kluczy (które mogą być liczbą lub dowolnym napisem).

34. Zaznacz zdanie prawdziwe

- (a) Architektura von Neumanna to sposób organizacji elementów komputera, w której dane przechowywane są w pamięci wspólnie z instrukcjami. <- tak
- (b) Architektura von Neumanna to sposób organizacji elementów komputera, w której dane przechowywane są w osobnej pamięci niż instrukcje.
- (c) Nie ma nic wspólnego z komputerami i określa styl architektoniczny z okresu II Wojny Światowej obowiązujący na terenach okupowanych przez Niemcy.

35. Procesor zapisuje 32-bitowe wartości w pamięci określając kolejności bajtów zgodnie z zasadą Big Endian. Wówczas wartość 4D3C2B1A umieszczona pod adresem 100 zajmie

- (a) adresy od 100 do 103 w kolejności 4D, 3C, 2B, 1A <- tak
- (b) adresy od 103 do 100 w kolejności 4D, 3C, 2B, 1A
- (c) adresy od 97 do 100 w kolejności 4D, 3C, 2B, 1A

36. Zaznacz zdanie prawdziwe

- (a) Unicode i UTF to dwa różne kreślenia na to samo.
- (b) Unicode i UTF różnią się tym, że Unicode jest 16-bitowe a UTF 32-bitowe.
- (c) Unicode i UTF to dwa ściśle powiązane, ale jednak różne pojęcia. <- tak

37. Zaznacz zdanie prawdziwe

- (a) Maszyna analityczna była pierwszym elektrycznym komputerem.
- (b) Maszyna analityczna to urządzenie mechaniczne powszechnie uważane za pierwszy programowalny komputer ogólnego zastosowania. <- tak
- (c) Maszyna analityczna to maszyna służąca do analitycznej analizy układów logicznych.