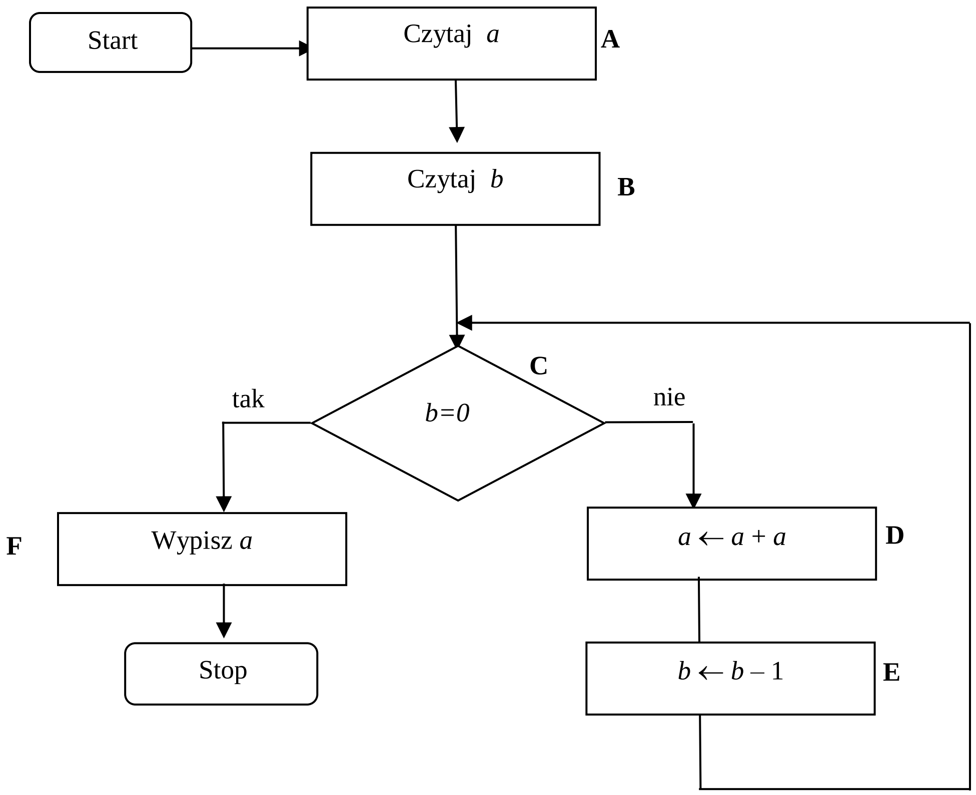
Imię i nazwisko: .... **WdI 2020/21, Kolokwium, IIn UWr**
Indeks:..

1. **[12 pkt]** Dany jest następujący algorytm w postaci schematu blokowego:



a) **[8]** Wykonaj polecenia:

   (a) Napisz program w kodzie RAM implementujący podany powyżej schemat blokowy. Twój program powinien być tak napisany, aby zmiennym a i b odpowiadały ustalone komórki pamięci oraz aby każdemu blokowi schematu odpowiadały ustalone instrukcje Twojego programu.
   (b) Podaj numery komórek pamięci odpowiadających zmiennym a i b.
   (c) Ponumeruj instrukcje Twojego programu kolejnymi liczbami naturalnymi i podaj, które instrukcje implementują poszczególne bloki (A – F) powyższego schematu:

b) **[4]** Uzupełnij specyfikację algorytmu.

Specyfikacja

Wejście: a, b – liczby całkowite nieujemne

Wyjście: ....

....

2. **[16 pkt]** Dana jest następująca funkcja:

```
funkcja zagadka(A, lw, pr)
    # A – tablica liczb całkowitych dodatnich
    # lw, pr - liczby całkowite nieujemne
    # x, y – tablice dwuelementowe
1. jeżeli (lw = pr):                zwróć( A[ lw ], –1 )
2. jeżeli (lw = pr – 1 ):
2a.     x[0] ← max( A[lw], A[pr] ), y[0] ← min( A[lw], A[pr] )
2b.     zwróć( x[0], y[0] )
3. s ← ( lw + pr ) // 2     # // - dzielenie całkowitoliczbowe, zaokrąglenie w dół
4. ( x[0], y[0] ) ← zagadka( A, lw, s )
5. ( x[1], y[1] ) ← zagadka( A, s + 1, pr )
6. jeżeli (x[0] > x[1]):            i ← 0
7. w przeciwnym przypadku:  i ← 1
8. z ← max( x[ 1 – i ], y[ i ] )
9. zwróć( x[ i ], z )
```

gdzie **max**(a, b) i **min**(a, b) to funkcja zwracająca odpowiednio większą i mniejszą z liczb a, b gdy $a \neq b$ oraz liczbę a gdy $a = b$.

a) **[6]** Rozważ uruchomienie zagadka(A, 0, 9), gdzie komórki A[0..5] są wypełnione cyframi Twojego numeru indeksu a komórki A[6..9] są wypełnione cyframi bieżącego roku 2021. Podaj:

   i) Wynik zwracany przez powyżej opisane wywołanie zagadka(A, 0, 9).

   ii) Liczbę wywołań rekurencyjnych funkcji zagadka w trakcie uruchomienia zagadka(A, 0, 9).

   iii) Wartości lw i pr dla wszystkich wywołań rekurencyjnych, w którym wartość argumentu pr jest równa 9.

b) **[3]** Zapisz w postaci zależności rekurencyjnej asymptotyczną złożoność czasową funkcji zagadka, gdzie argumentem jest rozważana liczba elementów tablicy $n = pr – lw + 1$. Możesz w swoim rozwiązaniu skorzystać z poniższej wskazówki ilustrującej niepełną postać tej zależności;

$T(1) = T(2) = 1$

$T(n) = T(\ldots\ldots) +$ .... gdy n jest nieparzyste

$T(n) =$ .... gdy n jest parzyste

c) **[4]** Załóżmy, że n jest naturalną potęgą dwójki czyli $n = 2^k$ dla nieujemnej całkowitej liczby k. Przy tym założeniu rozwiąż zależność rekurencyjną z punktu b) metodą podstawienia.

d) **[3]** Uzupełnij specyfikację funkcji zagadka.

Specyfikacja
Wejście: A – tablica liczb całkowitych dodatnich
lw, pr - liczby całkowite nieujemne

Wyjście: para liczb x, z taka, że ....

Imię i nazwisko: .... **WdI 2020/21, Kolokwium, IIn UWr**
Indeks: ....

3. **[11 pkt]** Napisz algorytm realizujący poniższą specyfikację:

Specyfikacja
Wejście:
s , $m_1,\dots m_6$, $c_1,\dots,c_4$ – liczby ze zbioru $\{0, 1\}$ reprezentujące znak, mantysę i cechę (w U2) liczby zapisanej w reprezentacji zmiennopozycyjnej.
Wyjście:
Wartość liczby o zapisie zmiennopozycyjnym

znak	mantysa						cecha			
s	m_1	m_2	m_3	m_4	m_5	m_6	c_1	c_2	c_3	c_4

Przykład:
Dla zapisu zmiennopozycyjnego

znak	mantysa						cecha			
0	0	1	0	0	0	0	1	1	1	1

Odpowiadającego ciągowi wejściowemu 0, 0, 1, 0, 0, 0, 0, 1, 1, 1, 1 wynik jest równy 0,625, gdyż $0{,}625 = (-1)^{0} \cdot 1{,}010000_{(2)} \cdot 2^{-1}$, czyli

- $s = 0$ odpowiada za znak dodatni liczby,
- $m_1,\dots, m_6 = 0, 1, 0, 0, 0, 0$ odpowiada mantysie `1,010000`
- $c_1,\dots, c_4 = 1, 1, 1, 1$ to wykładnik –1 zapisany w U2 na 4 bitach.

4. **[11 pkt]** Dla funkcji:

$$f_1(n) = n \cdot (\log n)^2$$

$$f_2(n) = n^2 / 10$$

$$f_3(n) = \log n + 2^{10} \cdot n$$

wykonaj następujące polecenia:

(a) Ustaw $f_1(n)$, $f_2(n)$ i $f_3(n)$ w takiej kolejności $f_i(n)$, $f_j(n)$, $f_k(n)$, że $f_i(n) = O(\, f_j(n)\,)$ oraz $f_j(n) = O(\, f_k(n)\,)$.

(b) Udowodnij każdą z zależności $f_i(n) = O(\, f_j(n)\,)$ oraz $f_j(n) = O(\, f_k(n)\,)$.

Uwaga. W dowodach zależności można m.in. korzystać z faktu:

Jeżeli $lim_{n\to\infty} \frac{f(n)}{g(n)} < \infty$, to $f(n) = O(\, g(n)\,)$.