

UZUPEŁNIA ZDAJĄCY			
KOD	PESEL	miejsce na naklejkę	

EGZAMIN MATURALNY Z INFORMATYKI

POZIOM ROZSZERZONY

Część I

DATA: **12 czerwca 2015 г.**

GODZINA ROZPOCZĘCIA: 9:00

CZAS PRACY: 60 minut

LICZBA PUNKTÓW DO UZYSKANIA: 15

MIN-R1 **1**P-153

UZUPEŁNIA ZDAJĄCY

WY	BRANE:	
(śroc	dowisko)	
(kor	mpilator)	
(prograi	m użytkowy)	

Instrukcja dla zdającego

- 1. Sprawdź, czy arkusz egzaminacyjny zawiera 7 stron. Ewentualny brak zgłoś przewodniczącemu zespołu nadzorującego egzamin.
- 2. Rozwiązania i odpowiedzi zamieść w miejscu na to przeznaczonym.
- 3. Pisz czytelnie. Używaj długopisu/pióra tylko z czarnym tuszem/atramentem.
- 4. Nie używaj korektora, a błędne zapisy wyraźnie przekreśl.
- 5. Pamiętaj, że zapisy w brudnopisie nie podlegają ocenie.
- 6. Wpisz zadeklarowane (wybrane) przez Ciebie na egzamin środowisko komputerowe, kompilator języka programowania oraz program użytkowy.
- 7. Jeżeli rozwiązaniem zadania lub jego części jest algorytm, to zapisz go w wybranej przez siebie notacji: listy kroków lub języka programowania, który wybrałaś/eś na egzamin.
- 8. Na tej stronie oraz na karcie odpowiedzi wpisz swój numer PESEL i przyklej naklejkę z kodem.
- 9. Nie wpisuj żadnych znaków w części przeznaczonej dla egzaminatora.

MIN 2015

Zadanie 1. Kod uzupełnień do dwóch

W kodzie uzupełnień do dwóch (w skrócie U2) zapisujemy liczby całkowite w dwójkowym systemie pozycyjnym. W n-bitowym systemie U2 reprezentujemy liczby całkowite z przedziału $[-(2^{n-1}), 2^{n-1} - 1]$. Przykładowo – dla n = 8 są to liczby:

Zapis $(a_{n-1}a_{n-2}...a_2a_1a_0)_{U2}$, gdzie $a_0, a_1, ..., a_{n-1} \in \{0,1\}$, oznacza liczbę

$$-a_{n-1} \cdot 2^{n-1} + a_{n-2} \cdot 2^{n-2} + \dots + a_1 \cdot 2^1 + a_0 \cdot 2^0$$
.

W porównaniu do klasycznego zapisu dwójkowego, w kodzie U2 najbardziej znaczący bit (a_{n-1}) ma przeciwną "wagę". Przykładowo: w reprezentacji 8-bitowej (n = 8) wagi bitów $a_0, a_1, a_2 a_3, a_4, a_5, a_6, a_7$ są równe odpowiednio: 1, 2, 4, 8, ..., 64, –128. W takim systemie reprezentacje liczb 117 i –82 wyglądają następująco:

$$(01110101)_{U2} = -0 \cdot 2^7 + 1 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 =$$

= 64 + 32 + 16 + 4 + 1 = 117

$$(10101110)_{U2} = -1 \cdot 2^7 + 0 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 =$$

$$= -128 + 32 + 8 + 4 + 2 = -82$$

Zadanie 1.1. (0-2)

Wykonaj poniższe polecenia:

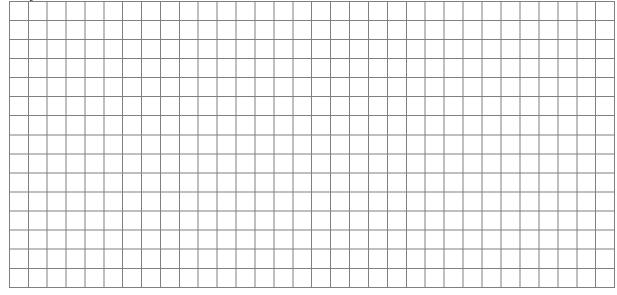
Zapisz liczby 93 i –42 w 8-bitowym kodzie U2:

$$93 = ($$
 $)_{U2}$ $-42 = ($ $)_{U2}$

Zapisz w 8-bitowym kodzie U2 wynik dodawania:

$$(00101011)_{U2} + (10011011)_{U2} = \dots$$

Miejsce na obliczenia:



Zadanie 1.2. (0-3)

Poniżej przedstawiono algorytm wyznaczania liczby przeciwnej do danej liczby zapisanej w kodzie U2.

Specyfikacja:

Dane:

liczba naturalna n > 1, reprezentacja $(a_{n-1} \ a_{n-2} \dots a_2 \ a_1 \ a_0)_{U2}$ liczby naturalnej $x \ (x \neq 0 \ \text{oraz} \ x \neq -2^{n-1})$ w n-bitowym kodzie U2.

Wynik:

reprezentacja $(b_{n-1} b_{n-2} \dots b_2 b_1 b_0)_{U2}$ liczby (-x) w n-bitowym kodzie U2.

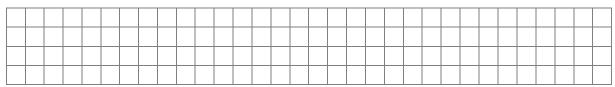
Algorytm:

- 1. $i \leftarrow 0$
- 2. dopóki $a_i = 0$ wykonuj:
 - a) $b_i \leftarrow 0$
 - b) $i \leftarrow i + 1$
- 3. $b_i \leftarrow a_i$
- 4. $i \leftarrow i + 1$
- 5. dopóki i < n wykonuj:
 - a) $b_i \leftarrow \mathbf{not}(a_i)$
 - b) $i \leftarrow i + 1$

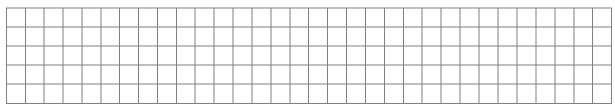
Uwaga:

not oznacza negację bitu, tzn. not(0) = 1, not(1) = 0.

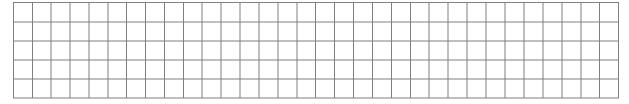
Podaj wynik wykonania algorytmu dla n = 16 i $x = (1111001001110000)_{U2}$



Podaj przykład liczby zapisanej w 16-bitowym kodzie U2, dla której algorytm nie wykona żadnej instrukcji z wnętrza pętli w **kroku 2**.



Podaj przykład liczby zapisanej w 16-bitowym kodzie U2, dla której algorytm w pętli z **kroku 5** wykona dokładnie 7 razy operację **not**.



Zadanie 2. Triady

Trzy dodatnie liczby a, b i c nazwiemy triadq, gdy możliwe jest utworzenie trójkąta, którego boki mają długości a, b i c.

Zadanie 2.1. (0-2)

Uzupełnij poniższe stwierdzenie.

Liczby dodatnie a, b i c spełniające warunek $a \le b$ tworzą triadę wtedy i tylko wtedy, gdy zachodzą jednocześnie następujące warunki:

Podaj, ile wartości c można dobrać ze zbioru

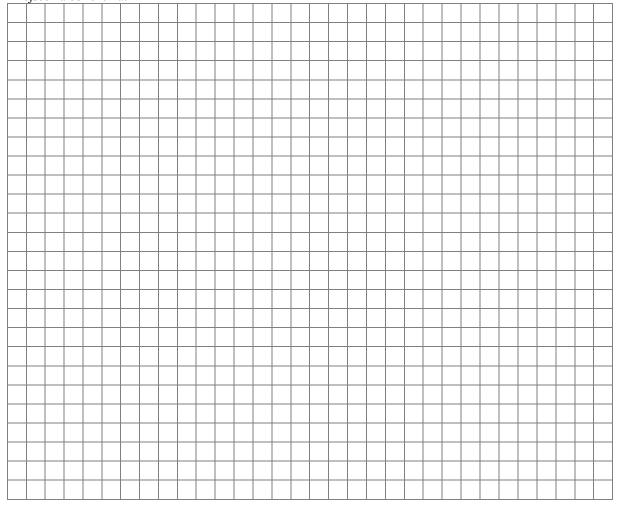
$$C = \{ 2, 3, 5, 6, 9, 10, 11, 13, 14, 15, 17, 19, 20, 23, 24 \}$$

tak, aby a = 5, b = 15 oraz $c \in C$ tworzyły triadę. Wskaż odpowiednie wartości c.

Elementy zbioru *C*, które wraz z *a* i *b* tworzą triadę:

Liczba elementów zbioru *C*, które wraz z *a* i *b* tworzą triadę:

Miejsce na obliczenia:



Zadanie 2.2. (0-4)

Zaproponuj algorytm, który dla całkowitego n \geq 2 wyznaczy wszystkie triady c_1 , c_2 i c_k , gdzie $1 \leq k \leq n$ (k może być równe 1 lub 2), w zadanym ciągu liczb $c_1, c_2, ..., c_n$. Swój algorytm zapisz zgodnie z poniższą specyfikacją.

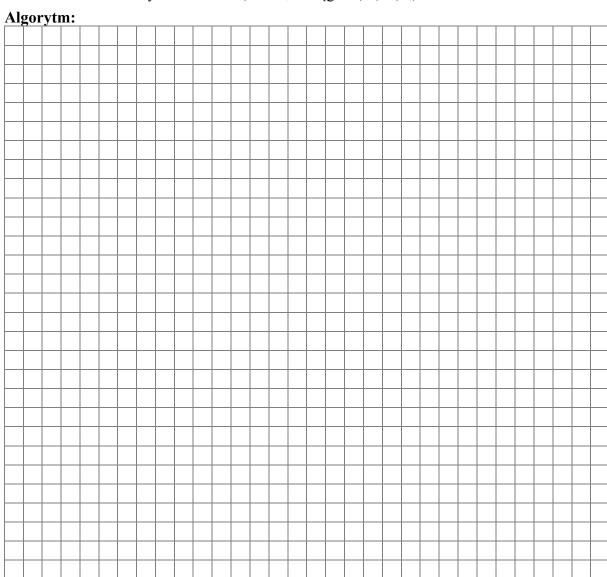
Specyfikacja:

Dane:

n- liczba elementów ciągu liczb, $n \ge 2$ c_1,c_2,\ldots,c_n- nieposortowany i bez powtarzających się elementów ciąg liczb dodatnich, w którym $c_1 < c_2$

Wynik:

liczba wszystkich triad c_1, c_2 i c_k w ciągu $c_1, c_2, ..., c_n, 1 \le k \le n$



Zadanie 3. Test

Oceń, czy poniższe zdania są prawdziwe. Zaznacz P, jeśli zdanie jest prawdziwe, albo F – jeśli zdanie jest fałszywe.

W każdym zadaniu punkt uzyskasz tylko za komplet poprawnych odpowiedzi.

Zadanie 3.1. (0-1)

Algorytm Euklidesa

1.	służy do obliczania potęgi a ^b .	P	F
2.	służy do obliczania największego wspólnego dzielnika dwóch liczb.	P	F
3.	zastosowany do liczb $a=100$, $b=10$ da wynik 5.	P	F
4.	zastosowany do liczb $a=100$, $b=8$ da wynik 4.	P	F

Zadanie 3.2. (0-1)

Liczba szesnastkowa (FCA)₁₆ jest

1.	mniejsza od liczby (FFF) ₁₆ .	P	F
2.	większa od liczby (AAAA) ₁₆ .	P	F
3.	mniejsza od liczby (1111) ₁₆ .	P	F
4.	większa od liczby (9999) ₁₆ .	P	F

Zadanie 3.3. (0-1)

Klucz obcy w tabeli bazy danych

1.	pochodzi z innej tabeli.	P	F
2.	służy do łączenia tabeli z inną tabelą.	P	F
3.	musi być opisany za pomocą jednej kolumny.	P	F
4.	jednoznacznie identyfikuje wiersze tej tabeli.	P	F

Zadanie 3.4. (0-1)

Adres IPv4

1.	składa się z 48-bitów.	P	F
2.	jest unikatowy w skali świata.	P	F
3.	jest unikatowy w skali sieci lokalnej.	P	F
4.	300.200.256.1 jest poprawny.	P	F

BRUDNOPIS (nie podlega ocenie)