## Zadanie 2.3. (0–2)

Podaj przykładową zawartość co najwyżej 10 elementowej tablicy $A$, dla której dla **każdego** $s = 1, 2, 3, \ldots, 200$ gra kończy się sukcesem.

Odpowiedź: _____

Miejsce na obliczenia:

## Zadanie 3. Potęgowanie modulo

Rozważmy operację potęgowania modularnego stosowaną np. w algorytmie RSA.

Liczbę $a$ podnosimy do potęgi $x$, po czym bierzemy resztę z dzielenia otrzymanej liczby przez ustaloną liczbę $M$, dzięki czemu otrzymujemy wynik

$$b = a^x \bmod M,$$

gdzie $a$, $M$ – dodatnie liczby całkowite, $x$ – nieujemna liczba całkowita.

Mówimy wtedy, że $a^x$ *modulo* $M$ *równa się* $b$.

**Przykład:**
Dla $a = 2$, $x = 5$, $M = 7$ liczymy resztę z dzielenia $2^5$ (czyli 32) przez 7, zatem $b = 4$.
Dla $a = 3$, $x = 3$ i $M = 11$ mamy $b = 3^3 \bmod 11 = 5$,
natomiast dla $a = 10$, $x = 2$ i $M = 13$ wynikiem jest $b = 10^2 \bmod 13 = 9$.

| Wypełnia egzaminator | Nr zadania | 2.1. | 2.2 | 2.3. |
|---|---|---|---|---|
| | Maks. liczba pkt. | 2 | 2 | 2 |
| | Uzyskana liczba pkt. | | | |

## Zadanie 3.1. (0–2)

Uzupełnij tabelę – podaj brakującą liczbę ($x$ lub $b$), dla której $a^x \bmod M = b$.

| M | a | x | b |
|---|---|---|---|
| 7 | 2 | 5 | 4 |
| 11 | 3 | 3 | |
| 31 | 5 | | 25 |
| 59 | 2 | | 5 |
| 80 | 9 | 2 | |

Miejsce na obliczenia:



## Zadanie 3.2. (0–4)

Zapisz w wybranej przez siebie notacji (w postaci pseudokodu lub w wybranym języku programowania) algorytm, który gdy są dane liczby $a$, $x$ i $M$, obliczy b = $a^x$ *mod M.* Aby otrzymać maksymalną liczbę punktów, Twój algorytm powinien wykonywać O(log $x$) operacji arytmetycznych wymienionych w poniższej uwadze.

**Uwaga**: W zapisie algorytmu możesz wykorzystać tylko operacje arytmetyczne: dodawanie, odejmowanie, mnożenie, dzielenie, dzielenie całkowite, resztę z dzielenia, oraz porównywanie liczb; instrukcje sterujące i przypisania do zmiennych lub samodzielnie napisane funkcje zawierające wyżej wymienione operacje.

**Specyfikacja:**

**Dane:**

*a* – liczba całkowita dodatnia

*x* – nieujemna liczba całkowita

*M* – liczba całkowita dodatnia

**Wynik:**

*b* – nieujemna liczba całkowita o wartości równej $a^x$ mod *M*

Algorytm:



| | | | |
|---|---|---|---|
| **Wypełnia egzaminator** | **Nr zadania** | **3.1.** | **3.2.** |
| | **Maks. liczba pkt.** | **2** | **4** |
| | **Uzyskana liczba pkt.** | | |

## Informacja do zadań 3.3.–3.5.

W pliku `liczby.txt` jest 1 000 wierszy, w każdym – po trzy nieujemne liczby całkowite, kolejno *M*, *a*, *b*, oddzielone pojedynczymi spacjami. Liczby w pliku są nie większe niż 10 000, a ponadto wszystkie liczby *M* i *a* są większe bądź równe 2.

Napisz **program**(-y), który(-e) znajdzie(-dą) odpowiedzi do poniższych zadań. Każdą odpowiedź zapisz w pliku `wyniki3.txt` i poprzedź ją numerem odpowiedniego zadania. Do Twojej dyspozycji jest plik `liczby_przyklad.txt`, w którym zapisano 5 wierszy w formacie opisanym wyżej. Odpowiedzi dla pliku przykładowego są podane przy odpowiednich zadaniach – możesz ich użyć, aby sprawdzić poprawność działania swojego programu.

### Zadanie 3.3. (0–2)

Oblicz, w ilu wierszach pliku `liczby.txt` liczba *M* jest liczbą pierwszą.

Dla pliku `liczby_przyklad.txt` odpowiedź wynosi 2.

### Zadanie 3.4. (0–2)

Oblicz, w ilu wierszach pliku `liczby.txt` pierwsze dwie zapisane liczby (*M* i *a*) są względnie pierwsze (to znaczy ich największym wspólnym dzielnikiem jest 1).

Dla pliku `liczby_przyklad.txt` odpowiedź wynosi 3.

### Zadanie 3.5. (0–2)

Dla każdej trójki liczb (*M, a, b*) zapisanej w jednym wierszu pliku rozstrzygnij, czy możliwe jest znalezienie takiego *x* z przedziału [0..*M* – 1], dla którego $a^x \bmod M = b$. Podaj, dla ilu trójek zachodzi taka sytuacja.

Dla pliku `liczby_przyklad.txt` odpowiedź wynosi 4.

**Do oceny oddajesz:**

- plik tekstowy `wyniki3.txt` zawierający odpowiedzi do poszczególnych zadań (odpowiedź do każdego zadania powinna być poprzedzona jego numerem)
- pliki zawierające kody źródłowe Twoich programów o nazwach odpowiednio:

  zadanie 3.3. .....

  zadanie 3.4. .....

  zadanie 3.5. .....