MATURA VS PROGRAMISTA

AGENDA:

- 18:30 Intro
- 18:40 Właściwy Start
- Koniec: około 22:00
 (zależy od tego, jak się wyrobię i ile będzie interakcji z Wami);

CO BĘDZIE?

- algorytmy;
- zadania z programowania;
- zadania zamknięte;

A w przyszłości:

- Access/Excel: w następnym streamie (jeśli się Wam spodoba);
- omówienie Waszych zadań;

Zasady na Streamie:

- 1. Smiało pytaj o co chcesz zadania z matur, sama matura, szkoła, programowanie, życie, etc...
- 2. Na każde pytanie postaram się odpowiedzieć:)
- 3. Daję sobie prawo do grzecznego odmówienia odpowiedzi
- 4. Apeluję o kulturę na czacie;)

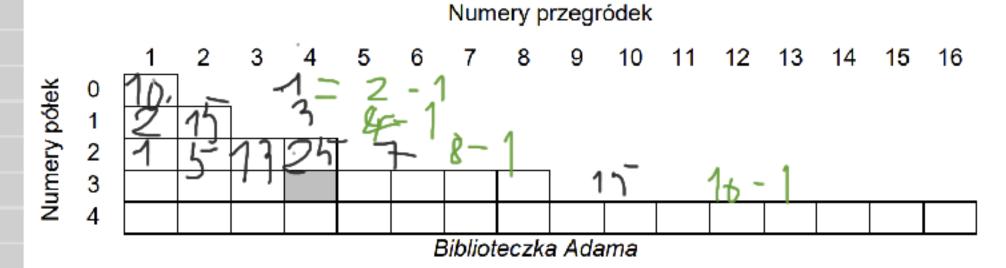
Zadanie 1.1.

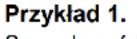
: 10, 2, 15, 13, 1, 5, 25

Przykład 1.

Szara komórka to przegródka B[3, 4]

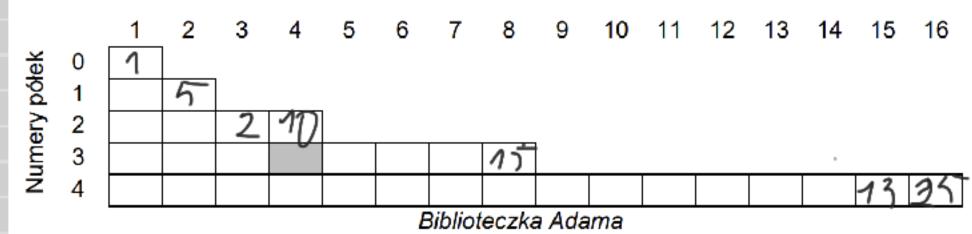
Adam sprawdza, czy przegródka B[i, j] ($i \ge 0$ oraz $1 \le j \le 2^i$) jest pusta. Jeśli tak, umieszcza książke w tej przegródce. W przeciwnym przypadku porównuje numer wstawianej książki z numerem książki w przegródce. Jeśli numer wstawianej książki jest mniejszy od numeru książki stojącej w przegródce, próbuje umieścić książkę na kolejnej półce w przegródce B[i + 1, 2j - 1]. Jeśli numer wstawianej książki jest większy od numeru książki w przegródce, to próbuje umieścić książkę w przegródce B[i + 1, 2j].

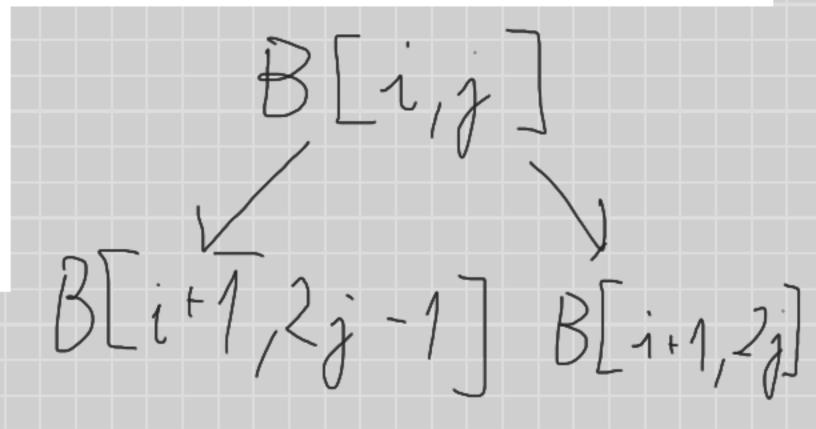




Szara komórka to przegródka *B*[3, 4] 1, 5, 10, 15, 2, 25, 13

Numery przegródek

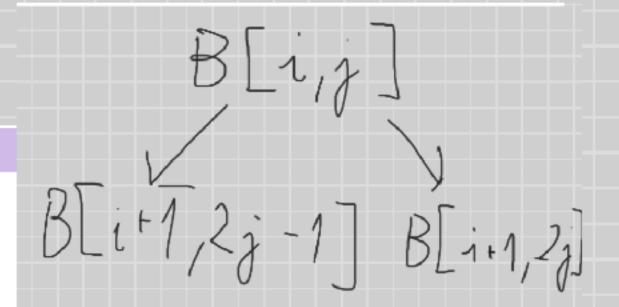




Zadanie 1.1

Zadanie 1.1. (0-2)

Podaj zawartość biblioteczki po wstawieniu do niej kolejno książek o numerach: 14, 18, 12, 9, 20, 15, 17.



Numery książek wpisz we właściwe miejsca na poniższym schemacie.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	14															
1	1,2	18)													
2	9		15	20												
3						17										
4																

Zadanie 1.2. (0-3)

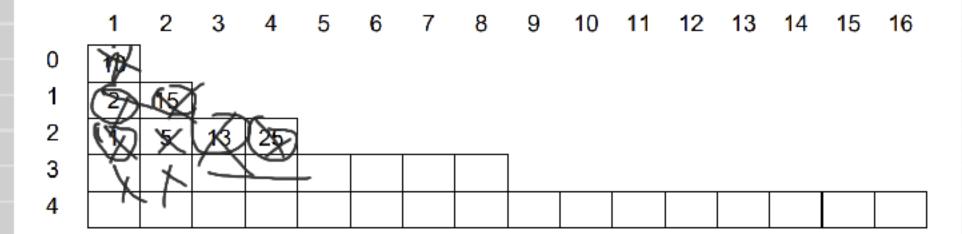
Uzupełnij tabelkę – wpisz, ile minimalnie, a ile maksymalnie musi być półek w biblioteczce, żeby można było umieścić w niej *n* książek i żeby na ostatniej półce znalazła się co najmniej jedna książka.

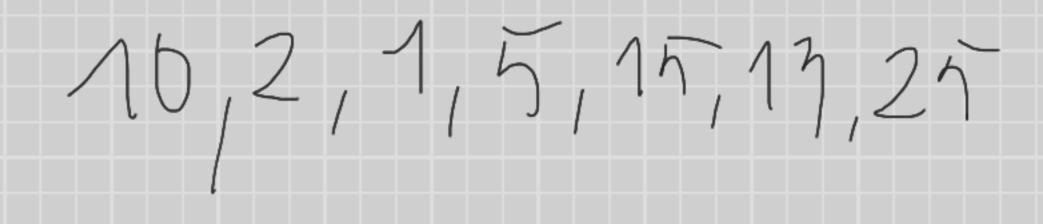
1	+2+1	- \$
t	16	

n – liczba książek	Minimalna liczba półek	Maksymalna liczba półek
1	1	1
3	2	3
4	3	4
7	3	7
16	5	16
31	7	31
32	6	32
$2^k - 1$, dla $k > 0$	K	2K-1

Przykład 2.

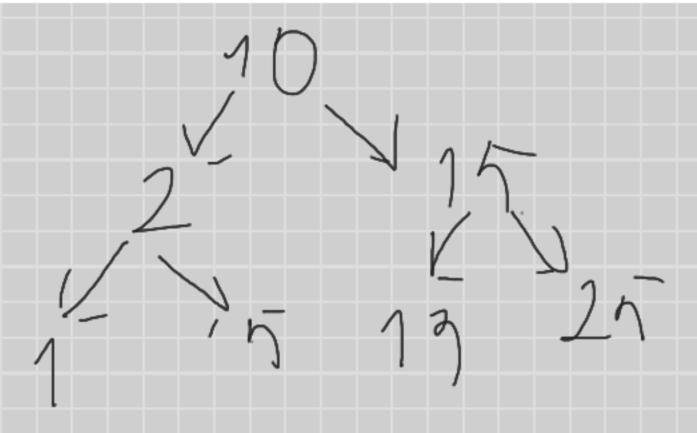
Poniżej przedstawiono zawartość biblioteczki po wstawieniu do niej książek kolejno o numerach: 10, 2, 15, 13, 1, 5, 25 (zakładamy, że przedtem biblioteczka była pusta).





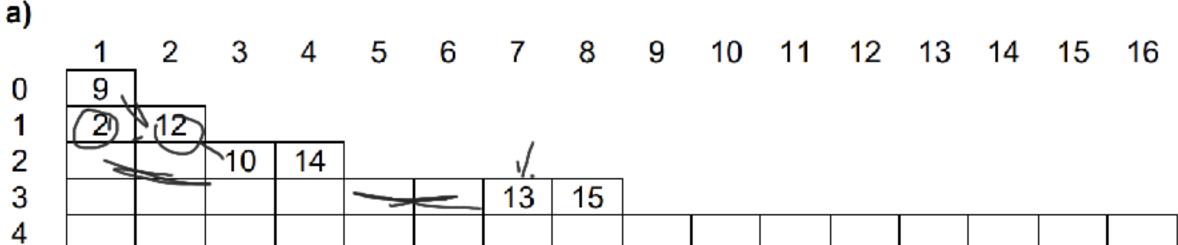
Dla biblioteczki z siedmioma książkami z przykładu 2. algorytm A wypisze: 10, 2, 1, 5, 15, 13, 25.

A(i, j)wypisz numer książki z przegródki B[i, j]jeżeli przegródka B[i + 1, 2j - 1] nie jest pusta, to
wykonaj A(i + 1, 2j - 1)jeżeli przegródka B[i + 1, 2j] nie jest pusta, to
wykonaj A(i + 1, 2j)



Zadanie 1.3.

Podaj ciągi liczb wypisane przez algorytm A dla podanych zawartości biblioteczki.



Odpowiedź: 0 2 12 10 14 17 17 15 b)

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 0 10 15 2 40 7 12 13 14 15 16

Odpowiedź: 10,8,4,6,15,12,13

A(i, j)
wypisz numer książki z przegródki B[i, j]
jeżeli przegródka B[i + 1, 2j - 1] nie jest pusta, to wykonaj A(i + 1, 2j - 1)
jeżeli przegródka B[i + 1, 2j] nie jest pusta, to wykonaj A(i + 1, 2j)

Rozwiązanie

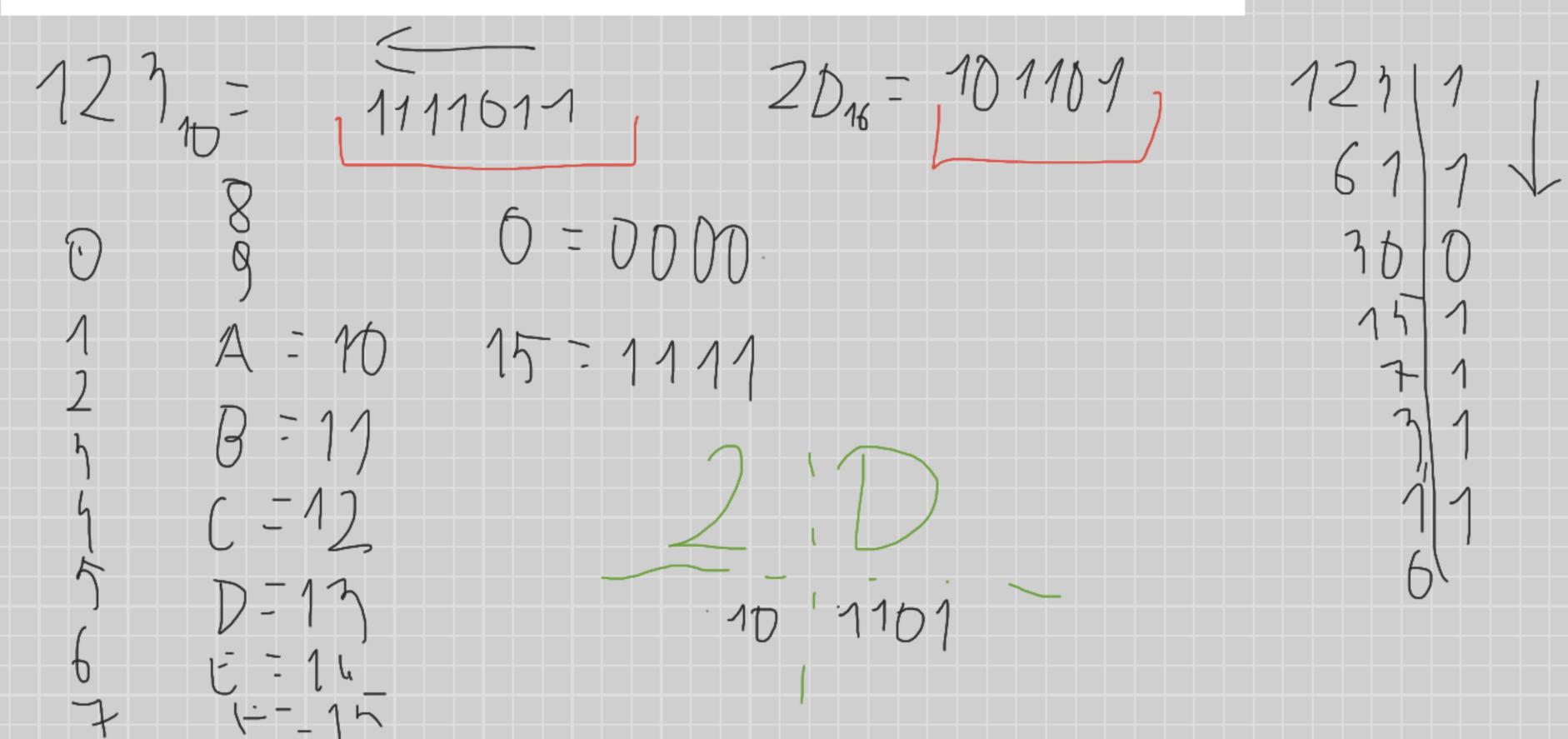
- a) 9, 2, 12, 10, 14, 13, 15
- b) 10, 8, 4, 6, 15, 12, 13

Zadanie 2.1

6432168 427

Zadanie 2.4.

Oblicz (123₁₀ XOR 101101₂) XOR 2D₁₆. Wynik podaj w systemie dziesiętnym.



Oblicz (123₁₀ XOR 101101₂) XOR 2D₁₆. Wynik podaj w systemie dziesiętnym.

$$-\frac{80}{-69} + 12 + 16 + 8 + 2 + 1 = 123$$

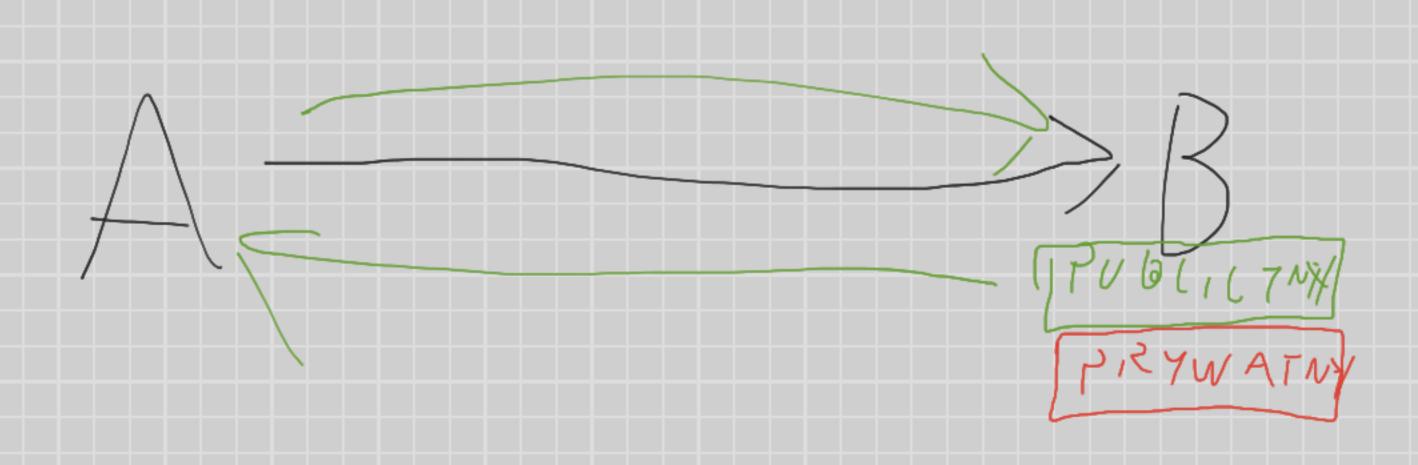
Zadanie 4. (0-1)

Oceń prawdziwość podanych zdań. Zaznacz **P**, jeśli zdanie jest prawdziwe, albo **F** – jeśli jest fałszywe.

4. 0–1

W komunikacji między dwoma osobami A i B z wykorzystaniem szyfrowania asymetrycznego klucz prywatny osoby A stosuje się do

1.	odszyfrowani <u>a w</u> iadomości wysłanej do osoby A przez osobę B.	(<u>P</u>)	F	
2.	uwierzytelnienia osoby B przez osobę A.	Р	F)



Zadanie 5. (0-1)

Dane są liczby zapisane w systemach pozycyjnych o podstawach 3, 5 i 6. Wstaw w miejsce kropek odpowiedni znak spośród: < , > , =, tak aby wyrażenie było poprawne.

$$(2011)_{3} = (134)_{6}$$

$$(134)_{6} - (13$$