

# Sprawozdanie

Daniel Majchrzycki, 218476

1. Program przypisuje wartość 1 do kolejnych komórek pamięci w tablicy dynamicznej, powiększając ją w przypadku braku miejsca. Program stosuje 3 strategie powiększania tablicy.

- a) Powiększanie tablicy o 1 w przypadku braku miejsca.
- b) Powiększanie tablicy o 100 w przypadku braku miejsca.
- c) Powiększanie tablicy o 10 000 w przypadku braku miejsca.

Testy zostały przeprowadzone dla różnych rozmiarów danych wejściowych:

- a)  $10^1$
- b)  $10^2$
- c)  $10^3$
- d)  $10^6$
- e)  $10^7$
- f)  $10^8$
- g)  $10^9$  – nie udało się prze symulować z powodu zbyt małej ilości pamięci RAM na testowym komputerze.

2. Wyniki – czasy podane w ms.

L.p	Strategia powiększania		
	1	100	10000
1	< 1	< 1	< 1
2	< 1	< 1	< 1
3	< 1	< 1	< 1
4	< 1	< 1	< 1
5	< 1	< 1	< 1
6	< 1	< 1	< 1
7	< 1	< 1	< 1
8	< 1	< 1	< 1
9	< 1	< 1	< 1
10	< 1	< 1	< 1
Średnia	< 1	< 1	< 1
Tablica $10^1$ elementowa			
1	< 1	< 1	< 1
2	< 1	< 1	< 1
3	< 1	< 1	< 1
4	< 1	< 1	< 1
5	< 1	< 1	< 1
6	< 1	< 1	< 1
7	< 1	< 1	< 1
8	< 1	< 1	< 1
9	< 1	< 1	< 1
10	< 1	< 1	< 1
Średnia	< 1	< 1	< 1
Tablica $10^2$ elementowa			
1	3	< 1	< 1
2	2	< 1	< 1
3	6	< 1	< 1
4	6	< 1	< 1

5	3	< 1	< 1
6	5	< 1	< 1
7	5	< 1	< 1
8	2	< 1	< 1
9	3	< 1	< 1
10	5	< 1	< 1
Średnia	4.2	< 1	< 1

Tablica 10<sup>3</sup> elementowa

	1	100	10000
1	130	5	< 1
2	135	6	< 1
3	134	6	< 1
4	132	4	< 1
5	131	5	< 1
6	135	3	< 1
7	125	5	< 1
8	134	6	< 1
9	133	5	< 1
10	123	5	< 1
Średnia	131.2	4.9	< 1

Tablica 10<sup>4</sup> elementowa

1	13 733	157	8
2	13 683	146	9
3	13 691	156	9
4	13 627	150	9
5	13 980	150	9
6	13 628	156	9
7	13 671	146	8
8	13 647	149	9
9	13 714	150	9
10	13 614	153	9
Średnia	13 699	151.3	8.8

Tablica 10<sup>5</sup> elementowa

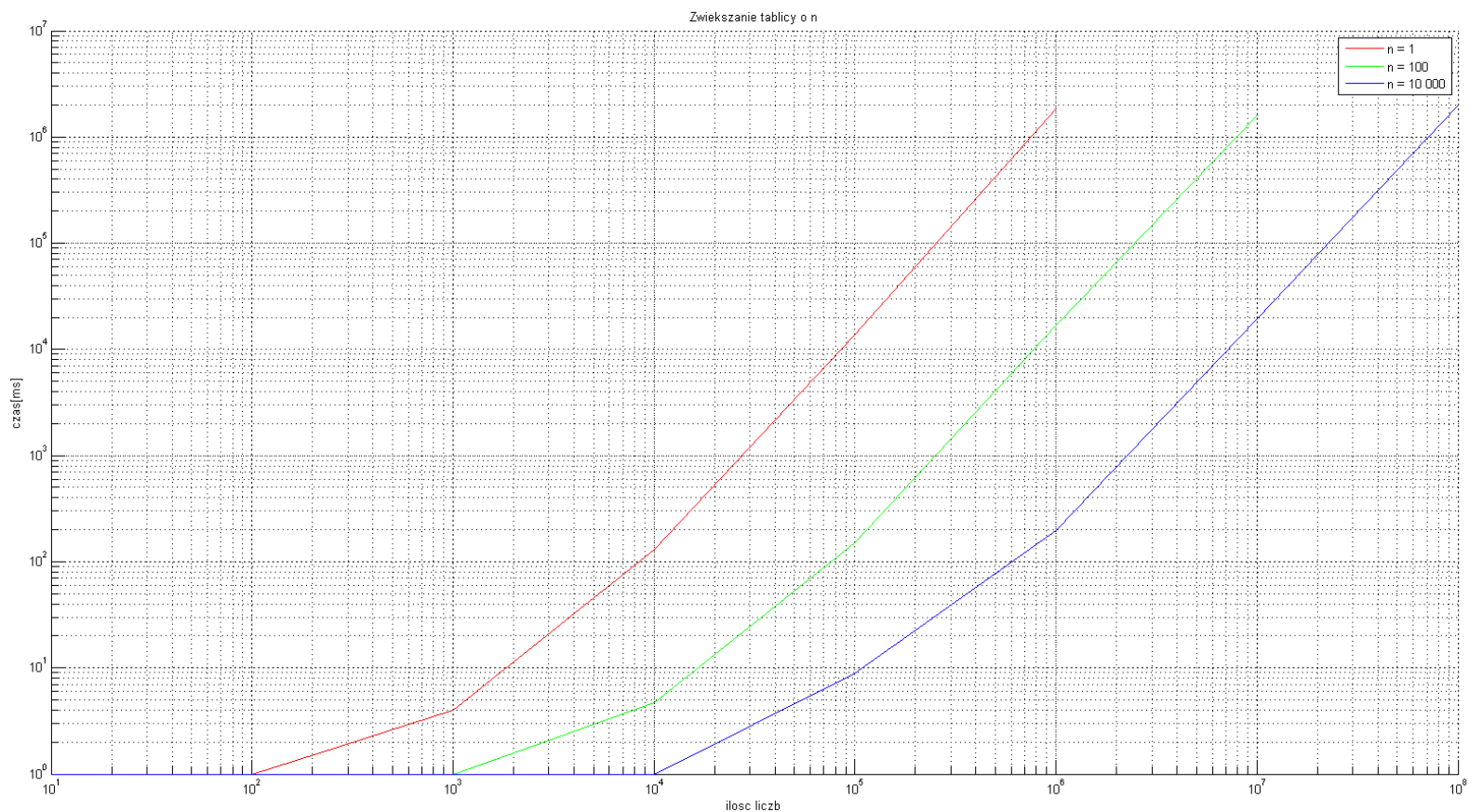
1	$1.884 \cdot 10^6$	16 111	197
2	$1.799 \cdot 10^6$	16 193	199
3	$1.870 \cdot 10^6$	16 715	194
4	$1.843 \cdot 10^6$	16 364	191
5	$1.821 \cdot 10^6$	17 269	196
6	$1.890 \cdot 10^6$	17 475	200
7	$1.901 \cdot 10^6$	17 194	193
8	$1.867 \cdot 10^6$	17 191	191
9	$1.852 \cdot 10^6$	17 267	200
10	$1.812 \cdot 10^6$	17 486	193
Średnia	$1.854 \cdot 10^6$	16 927	195.4

Tablica 10<sup>6</sup> elementowa

	1	100	10000
1	—	$1.602 \cdot 10^6$	16 965
2	—	$1.593 \cdot 10^6$	17 142
3	—	$1.586 \cdot 10^6$	17 257
4	—	$1.604 \cdot 10^6$	20 019
5	—	$1.642 \cdot 10^6$	20 985

6	—	$1.612 \cdot 10^6$	19 632
7	—	$1.630 \cdot 10^6$	20 632
8	—	$1.628 \cdot 10^6$	20 506
9	—	$1.597 \cdot 10^6$	20 251
10	—	$1.621 \cdot 10^6$	20 619
Średnia	—	$1.612 \cdot 10^6$	19 421
Tablica $10^7$ elementowa			
1	—	—	$1.957 \cdot 10^6$
2	—	—	$1.924 \cdot 10^6$
3	—	—	$1.974 \cdot 10^6$
4	—	—	$1.912 \cdot 10^6$
5	—	—	$1.948 \cdot 10^6$
6	—	—	$1.963 \cdot 10^6$
7	—	—	$1.981 \cdot 10^6$
8	—	—	$1.962 \cdot 10^6$
9	—	—	$1.879 \cdot 10^6$
10	—	—	$1.917 \cdot 10^6$
Średnia	—	—	$1.9417 \cdot 10^6$
Tablica $10^8$ elementowa			

### 3. Wykres



### 4. Wnioski

Algorytm zwiększania tablicy o 1 jest bezużyteczny w przypadku dużych tablic, z powodu bardzo dużej ilości operacji kopiowania całej tablicy.

Algorytm zwiększania tablicy o 100 osiąga znacznie lepszy rezultat, nie powodując przy tym sytuacji w której duża część zaalokowanej pamięci jest nie używana.

Algorytm zwiększania tablicy o 10 000 oferuje największą wydajność i może być płynnie używany dla ilości liczb rzędu  $10^6$ .

Dla każdej strategii powiększania tablicy istnieje granica dla której złożoność obliczeniowa zaczyna gwałtownie (wykładniczo) rosnąć. Granica ta zbliżona jest do wartości o którą powiększana jest tablica