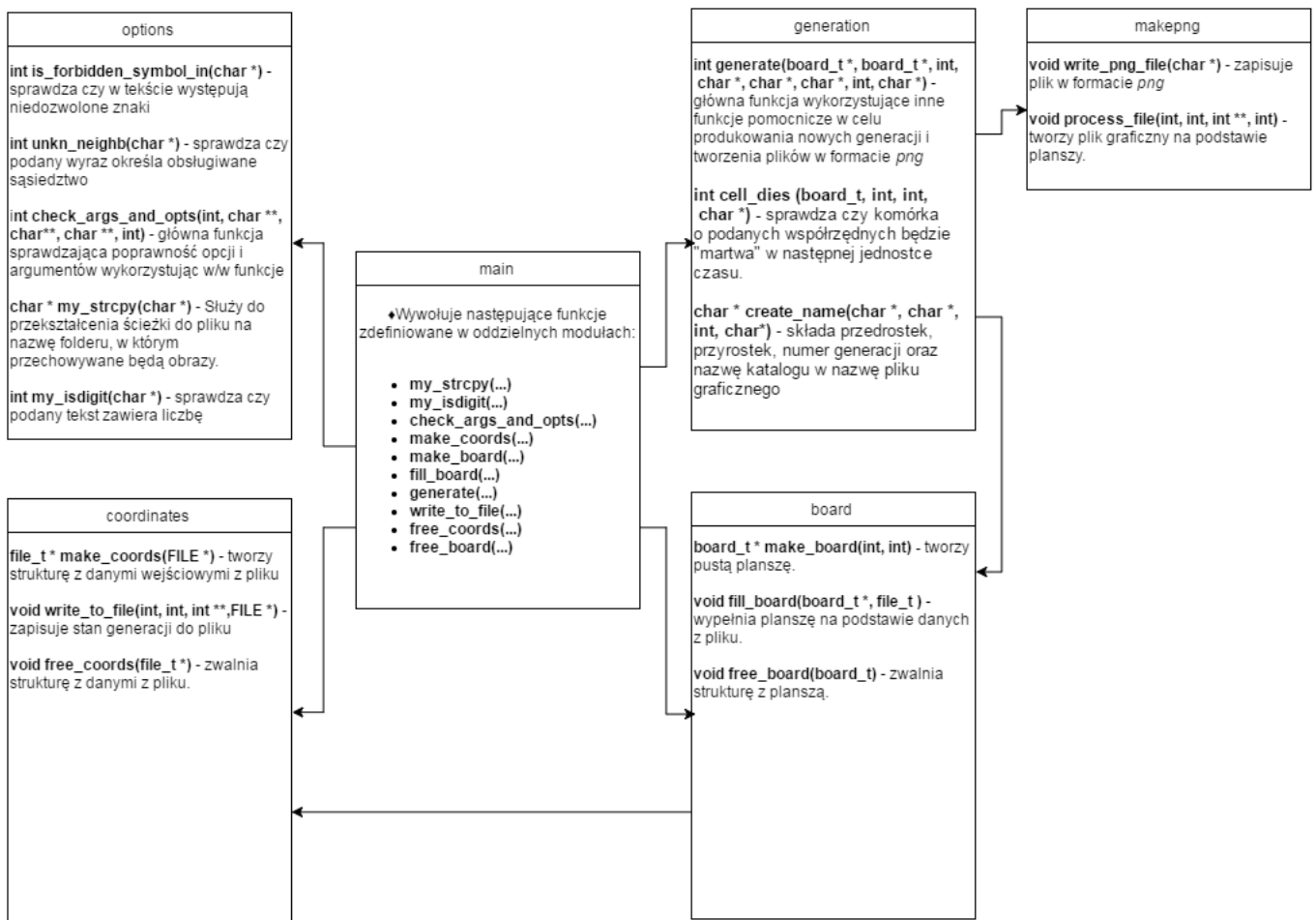


Sprawozdanie końcowe

Michał Sut

1 Ostateczna implementacja projektu

1.1 Diagram modułów



1.2 Opis modułów

1.2.1 main

Moduł główny. Odpowiada za pobranie argumentów i opcji, oraz obsługę błędów zgłaszanych przez funkcje. Wywoływane są w nim zasadnicze funkcje programu, takie jak tworzenie struktur danych czy przeprowadzenie generacji. W tym celu **main** odwołuje się do modułów: **options**, **coordinates**, **board** oraz **generation**.

1.2.2 options

Zawiera funkcje sprawdzające poprawność podanych opcji uruchomienia. Wartość zwracana przez funkcję to typ wyliczeniowy. Możliwe zwracane wartości:

ALL_CORRECT - Wszystkie wprowadzone opcje poprawne.

N_NEGATIVE - Ujemna liczba generacji.

UNKN_NEIGHB - Nieznany typ sąsiedztwa.

SCALE_NEGATIVE - Ujemne powiększenie.

FORBIDDEN_SYMBOLS - Przedrostek lub przyrostek zawierają nieprawidłowe znaki.

1.2.3 coordinates

Przechowuje strukturę zawierającą dane z pliku wejściowego oraz dane, które będą wypisane do pliku wyjściowego. Zawiera także funkcje operujące na tej strukturze: stworzenie struktury (z jednoczesnym pobraniem informacji z pliku) oraz wypisanie do pliku.

1.2.4 board

Moduł realizujący obsługę planszy na której toczy się gra. Zdefiniowane są w nim: struktura planszy, funkcja tworząca, usuwająca i wypełniająca tę strukturę. Wiąże się z: **coordinates**, skąd czerpie informacje niezbędne do wypełnienia planszy.

1.2.5 generate

Najważniejsza część programu. Realizuje **n** generacji oraz każdą z nich zapisuje do pliku ***.png**. Powiązana jest z modulem **board** z którego pobiera planszę, oraz **makepng**, z którego wykorzystuje funkcję zapisu do pliku graficznego.

1.2.6 makepng

Dodatkowy fragment programu stworzony dla modułu **generate** zawierający funkcję zapisującą strukturę planszy do pliku w formacie png.

1.3 Scenariusz działania programu

1. Odczytanie informacji z argumentów wywołania oraz opcji uruchomienia.
2. Sprawdzenie poprawności opcji i argumentów
3. Utworzenie struktury **file_t** na podstawie danych z pliku wejściowego
4. Utworzenie, a następnie wypełnienie struktury **board_t** z planszą na podstawie informacji z **file_t**.
5. Przeprowadzenie generacji z jednoczesnym zapisem do plików graficznych.
6. Wypisanie danych ostatniej generacji do pliku wyjściowego
7. Zwolnienie pamięci zajmowanych przez struktury

2 Testowanie działania programu

2.1 Poprawne dane, poprawne uruchomienie

Poprawne dane:

Dwie pierwsze liczby oznaczają odpowiednio szerokość i wysokość planszy, następnie pary liczb x y będące współrzędnymi żywych komórek. Dane nie zawierają liczb ujemnych oraz znaków innych niż cyfry. Liczba par x y jest nie większa niż szerokość*wysokość

Poprawne uruchomienie:

```
./life [wejście] [wyjście] [-p tekst] [-q tekst] [-n liczba] [-s tekst] [-x liczba]
```

[wejście] – nazwa istniejącego pliku z danymi wejściowymi. Brak argumentu oznacza ręczne wprowadzanie danych.

[wyjście] – nazwa pliku, do którego ma być zapisana ostatnia generacja. Brak argumentu oznacza wypisanie danych w oknie konsoli.

[-p] – przedrostek nazw plików graficznych. Brak tej opcji oznacza przedrostek domyślny "picture"

[-q] – przyrostek nazw plików graficznych. Brak tej opcji oznacza brak przyrostka

[-n] – liczba żądanych generacji. Brak oznacza wygenerowanie tylko jednego obrazu zawierającego stan początkowy.

[-s] – sąsiedztwo wykorzystywane w generacjach. Brak opcji oznacza sąsiedztwo domyślne: "Moore"

[-x] – powiększenie obrazu. Brak opcji oznacza brak powiększenia

Wynik:

Informacja o pomyślnym wygenerowaniu n obrazów.

Szczegółowy przebieg działania programu został przedstawiony w punkcie 5. *Prezentacja działania aplikacji.*

2.2 Błędne uruchomienie

Błędne uruchomienie:

Uruchomienie uznawane jest za błędne, gdy nie istnieje plik wejściowy o podanej nazwie lub gdy co najmniej jedna opcja otrzyma błędny argument. Niepoprawne argumenty opcji to :

[-p] – tekst zawierający znaki: \ / : * ? | < > "

[-q] – tekst zawierający znaki: \ / : * ? | < > "

[-n] – tekst zawierający znaki inne niż cyfry lub liczba ujemna.

[-s] – tekst inny niż "moore" lub "neumann" (wielkość liter nie ma znaczenia).

[-x] – tekst zawierający znaki inne niż cyfry lub liczba ujemna.

Przykładowe uruchomienia i wyniki:

1. Nie istniejący plik wejściowy

Polecenie: `./life wrongfile Output -p img -n 10 -s moore -x 4`

Wynik: `4 - No such file: wrongfile`

2. Ujemna liczba po opcji -n

Polecenie: `./life InputData Output -p img -n -12 -s moore -x 4`

Wynik: `3 - Argument of '-n' option must be a positive number[given: -n -12]`

3. Znaki inne niż cyfry po opcji -n

Polecenie: `./life InputData Output -p img -n aa -s moore -x 4`

Wynik: 3 - Option '-n' requires a number. [given: aa]

4. Niedozwolone znaki po opcji -p (Dla opcji -q wynik identyczny)

Polecenie: ./life InputData Output -p tex/t -n 10 -s moore -x 4

Wynik:

3 - Prefix or suffix cannot contain any of the following characters: \/:*?| < > "

5. Nieznane sąsiedztwo

Polecenie: ./life InputData Output -p img -n 10 -s newton -x 4

Wynik: 3 - Unknown neighbourhood: "newton".

Avaiable: "Moore", "Neumann" (capitalization doesn't matter).

6. Znaki inne niż cyfry po opcji -x

Polecenie: ./life InputData Output -p img -n 10 -s moore -x b4

Wynik: 3 - Option '-x' requires a number. [given: b4]

7. Ujemna liczba po opcji -x

Polecenie: ./life InputData Output -p img -n -10 -s moore -x -5

Wynik: 3 - Argument of '-x' option must be a positive number[given: x = -5]

Example: -x 10 makes generations 10 times larger

2.3 Błędne dane

Dane są błędne jeśli zawierają liczby ujemne i\lub znaki inne niż cyfry, a także jeśli liczba par jest większa niż ilość pól planszy lub gdy para jest nie pełna (tzn. dana jest tylko jedna współrzędna)

Błędne dane:

3

3

2 xyz

1 1

Wynik:

2- Can't load data from file "test". It's corrupt or contains wrong data format.

The input data should be positive integers.

The first two numbers are respectively: width and height. Then, there are pairs "x y".

'x' must be greater than 0 and less than or equal to 'width'.

'y' must be greater than 0 and less than or equal to 'height'.

Powyższy komunikat wyświetlany jest dla dowolnych błędnych danych.

2.4 Błędne dane i błędne uruchomienie

Jeśli podane zostaną jednocześnie błędne dane oraz niepoprawne argumenty opcji, zostanie wyświetlony komunikat informujący o niepoprawności pierwszej ze sprawdzonych opcji.

3 Testy modułów

Testy przeprowadzane będą dla następujących danych:

```
45
30
22 14
23 12
23 14
25 13
26 14
27 14
28 14
```

3.1 coordinates

Test tego modułu polega na sprawdzeniu, czy wszystkie informacje z pliku zostaną przesłane do struktury `file_t`. Sprawdzone zostanie, czy program prawidłowo odczytał szerokość i wysokość planszy, współrzędne punktów oraz czy zliczona ilość punktów jest taka sama jak ilość punktów w pliku testowym. Poniższy zrzut ekranu pokazuje, że moduł ten działa poprawnie.

```
sutm@jimp:~/jimp2/life2$ cat test_data
45
30
22 14
23 12
23 14
25 13
26 14
27 14
28 14
sutm@jimp:~/jimp2/life2$ ./testCoordinates test_data
Width: 45
Height: 30
Number of pairs: 7
Pairs:
x=22 y=14
x=23 y=12
x=23 y=14
x=25 y=13
x=26 y=14
x=27 y=14
x=28 y=14
sutm@jimp:~/jimp2/life2$
```

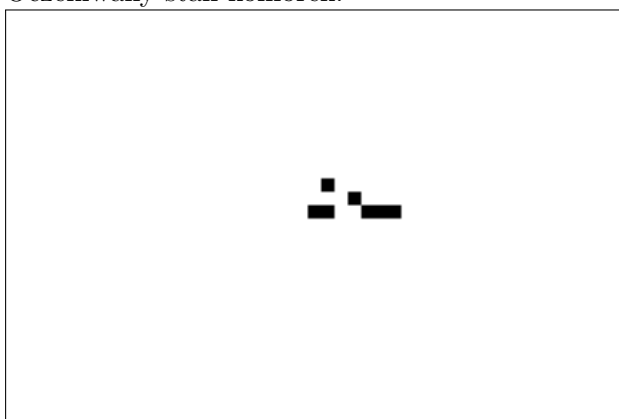
3.2 board

Test modułu **board** polega na sprawdzeniu, czy współrzędne żywych komórek zostały prawidłowo przekształcone na komórki planszy. Jeśli tak, plansza powinna zostać wypełniona zerami i jedynkami, przy czym 1 oznacza żywą komórkę. W tym celu generacja początkowa zostanie wypisana w oknie konsoli i porównana z jej graficznym przedstawieniem. Poniższy zrzut ekranu pokazuje, że struktura **board** została wypełniona zgodnie z oczekiwaniami.

Wynik testu:

[illegible]

Oczekiwany stan komórek:



3.3 options

Test tego modułu polegałby na sprawdzeniu jak reaguje program na błędne opcje i argumenty. To zostało jednak pokazane w punkcie 2. *Testowanie działania programu*, gdzie zbadano zachowanie programu dla każdego błędnego uruchomienia. Program zwracał odpowiednie komunikaty, co pozwala uznać test za zaliczony.

3.4 generation, makepng

Moduł **generation** przetestowany został pod względem tego, czy kolejne generacje tworzone są zgodnie z zasadami *Gry w życie* stworzonymi przez Johna Conway’a. **makepng** powinien stworzyć obrazy odpowiadające stanom planszy. Aby sprawdzić poprawność tych modułów, dwie kolejne generacje wypisane na ekran oraz stworzone obrazy zostaną porównane z dwiema kolejnymi generacjami przeprowadzonymi przez zewnętrzną aplikację([Gra w życie - online](#)). Poniższe zrzuty ekranu pokazują, że moduł działa prawidłowo.

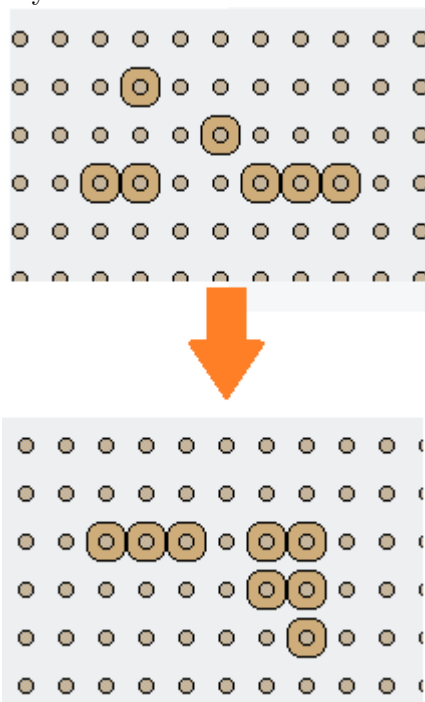
Wyniki testu:

[illegible]

Wygenerowane obrazy:



Wyniki oczekiwane:



4 Sprawdzenie wycieków pamięci programem "Valgrind"

Program został przetestowany pod kątem zajmowanej i zwalnianej pamięci. Niestety, pomimo użycia funkcji usuwających wykorzystywane struktury, nie udało się całkowicie wyeliminować problemu "wycieków" pamięci. Jest ona tracona w funkcjach odpowiedzialnych za tworzenie ciągów znaków lub struktur. Jednakże, znacznie większa część jest tracona podczas tworzenia plików graficznych **.png*. Spowodowane jest to głównie przez nieuwalnianie wektorów przechowujących informacje o pliku. Poniżej zaprezentowano zrzut ekranu z działania programu **Valgrind**, który bada wycieki pamięci. Wyniki dotyczą wygenerowania 100 obrazów o rozmiarze 45 x 30, powiększonych czterokrotnie.


```

sutm@jimp: ~/jimp2/life2
sutm@jimp:~/jimp2/life2$ valgrind --leak-check=full ./life data/Acorn data/AcornOutp -n100 -x4
==31913== Memcheck, a memory error detector
==31913== Copyright (C) 2002-2013, and GNU GPL'd, by Julian Seward et al.
==31913== Using Valgrind-3.10.0.SVN and LibVEX; rerun with -h for copyright info
==31913== Command: ./life data/Acorn data/AcornOutp -n100 -x4
==31913==
Successfully generated 100 pictures.
==31913==
==31913== HEAP SUMMARY:
==31913==    in use at exit: 28,195,848 bytes in 2,031 blocks
==31913==   total heap usage: 14,452 allocs, 12,421 frees, 30,547,648 bytes allocated
==31913==
==31913== 1 bytes in 1 blocks are definitely lost in loss record 1 of 46
==31913==    at 0x4C2AB80: malloc (in /usr/lib/valgrind/vgpreload_memcheck-amd64-linux.so)
==31913==    by 0x401EA5: check_args_and_opts (in /home/sutm/jimp2/life2/life)
==31913==    by 0x4013E4: main (in /home/sutm/jimp2/life2/life)
==31913==
==31913== 6 bytes in 1 blocks are definitely lost in loss record 2 of 46
==31913==    at 0x4C2AB80: malloc (in /usr/lib/valgrind/vgpreload_memcheck-amd64-linux.so)
==31913==    by 0x401EFC: check_args_and_opts (in /home/sutm/jimp2/life2/life)
==31913==    by 0x4013E4: main (in /home/sutm/jimp2/life2/life)
==31913==
==31913== 8 bytes in 1 blocks are definitely lost in loss record 3 of 46
==31913==    at 0x4C2AB80: malloc (in /usr/lib/valgrind/vgpreload_memcheck-amd64-linux.so)
==31913==    by 0x401E71: check_args_and_opts (in /home/sutm/jimp2/life2/life)
==31913==    by 0x4013E4: main (in /home/sutm/jimp2/life2/life)
==31913==
==31913== 11 bytes in 1 blocks are definitely lost in loss record 6 of 46
==31913==    at 0x4C2AB80: malloc (in /usr/lib/valgrind/vgpreload_memcheck-amd64-linux.so)
==31913==    by 0x40117D: main (in /home/sutm/jimp2/life2/life)
==31913==
==31913== 14 bytes in 1 blocks are definitely lost in loss record 7 of 46
==31913==    at 0x4C2AB80: malloc (in /usr/lib/valgrind/vgpreload_memcheck-amd64-linux.so)
==31913==    by 0x401F6E: my_strcpy (in /home/sutm/jimp2/life2/life)
==31913==    by 0x4011D7: main (in /home/sutm/jimp2/life2/life)
==31913==
==31913== 15 bytes in 1 blocks are definitely lost in loss record 8 of 46
==31913==    at 0x4C2AB80: malloc (in /usr/lib/valgrind/vgpreload_memcheck-amd64-linux.so)
==31913==    by 0x4011C0: main (in /home/sutm/jimp2/life2/life)
==31913==
==31913== 20 bytes in 1 blocks are definitely lost in loss record 9 of 46
==31913==    at 0x4C2AB80: malloc (in /usr/lib/valgrind/vgpreload_memcheck-amd64-linux.so)
==31913==    by 0x4020D3: generate (in /home/sutm/jimp2/life2/life)
==31913==    by 0x4015F3: main (in /home/sutm/jimp2/life2/life)
==31913==
==31913== 256 bytes in 2 blocks are definitely lost in loss record 16 of 46
==31913==    at 0x4C2AB80: malloc (in /usr/lib/valgrind/vgpreload_memcheck-amd64-linux.so)
==31913==    by 0x401AA4: make_board (in /home/sutm/jimp2/life2/life)
==31913==    by 0x401519: main (in /home/sutm/jimp2/life2/life)
==31913==
==31913== 256 bytes in 2 blocks are definitely lost in loss record 17 of 46
==31913==    at 0x4C2AB80: malloc (in /usr/lib/valgrind/vgpreload_memcheck-amd64-linux.so)

```

```

sutm@jimp: ~/jimp2/life2
==31913==    by 0x401AA4: make_board (in /home/sutm/jimp2/life2/life)
==31913==    by 0x401519: main (in /home/sutm/jimp2/life2/life)
==31913==
==31913== 256 bytes in 2 blocks are definitely lost in loss record 17 of 46
==31913==    at 0x4C2AB80: malloc (in /usr/lib/valgrind/vgpreload_memcheck-amd64-linux.so)
==31913==    by 0x401AA4: make_board (in /home/sutm/jimp2/life2/life)
==31913==    by 0x40153A: main (in /home/sutm/jimp2/life2/life)
==31913==
==31913== 505 bytes in 101 blocks are definitely lost in loss record 19 of 46
==31913==    at 0x4C2AB80: malloc (in /usr/lib/valgrind/vgpreload_memcheck-amd64-linux.so)
==31913==    by 0x4024B2: create_name (in /home/sutm/jimp2/life2/life)
==31913==    by 0x402157: generate (in /home/sutm/jimp2/life2/life)
==31913==    by 0x4015F3: main (in /home/sutm/jimp2/life2/life)
==31913==
==31913== 505 bytes in 101 blocks are definitely lost in loss record 20 of 46
==31913==    at 0x4C2AB80: malloc (in /usr/lib/valgrind/vgpreload_memcheck-amd64-linux.so)
==31913==    by 0x4024C0: create_name (in /home/sutm/jimp2/life2/life)
==31913==    by 0x402157: generate (in /home/sutm/jimp2/life2/life)
==31913==    by 0x4015F3: main (in /home/sutm/jimp2/life2/life)
==31913==
==31913== 1,010 bytes in 101 blocks are definitely lost in loss record 23 of 46
==31913==    at 0x4C2AB80: malloc (in /usr/lib/valgrind/vgpreload_memcheck-amd64-linux.so)
==31913==    by 0x4024EE: create_name (in /home/sutm/jimp2/life2/life)
==31913==    by 0x402157: generate (in /home/sutm/jimp2/life2/life)
==31913==    by 0x4015F3: main (in /home/sutm/jimp2/life2/life)
==31913==
==31913== 3,535 bytes in 101 blocks are definitely lost in loss record 25 of 46
==31913==    at 0x4C2AB80: malloc (in /usr/lib/valgrind/vgpreload_memcheck-amd64-linux.so)
==31913==    by 0x4025C7: create_name (in /home/sutm/jimp2/life2/life)
==31913==    by 0x402157: generate (in /home/sutm/jimp2/life2/life)
==31913==    by 0x4015F3: main (in /home/sutm/jimp2/life2/life)
==31913==
==31913== 46,400 bytes in 100 blocks are definitely lost in loss record 34 of 46
==31913==    at 0x4C2AB80: malloc (in /usr/lib/valgrind/vgpreload_memcheck-amd64-linux.so)
==31913==    by 0x4E51F01: ??? (in /lib/x86_64-linux-gnu/libpng12.so.0.50.0)
==31913==    by 0x4E3AEA0: png_create_info_struct (in /lib/x86_64-linux-gnu/libpng12.so.0.50.0)
==31913==    by 0x402733: write_png_file (in /home/sutm/jimp2/life2/life)
==31913==    by 0x402186: generate (in /home/sutm/jimp2/life2/life)
==31913==    by 0x4015F3: main (in /home/sutm/jimp2/life2/life)
==31913==
==31913== 65,536 bytes in 1 blocks are possibly lost in loss record 39 of 46
==31913==    at 0x4C2AB80: malloc (in /usr/lib/valgrind/vgpreload_memcheck-amd64-linux.so)
==31913==    by 0x4E5202C: png_malloc (in /lib/x86_64-linux-gnu/libpng12.so.0.50.0)
==31913==    by 0x4E3ACED: ??? (in /lib/x86_64-linux-gnu/libpng12.so.0.50.0)
==31913==    by 0x5427F5B: deflateInit2_ (in /lib/x86_64-linux-gnu/libz.so.1.2.8)
==31913==    by 0x4E4361D: ??? (in /lib/x86_64-linux-gnu/libpng12.so.0.50.0)
==31913==    by 0x4E4932A: png_write_info_before_PLTE (in /lib/x86_64-linux-gnu/libpng12.so.0.50.0)
==31913==    by 0x4E4954C: png_write_info (in /lib/x86_64-linux-gnu/libpng12.so.0.50.0)
==31913==    by 0x402822: write_png_file (in /home/sutm/jimp2/life2/life)
==31913==    by 0x402186: generate (in /home/sutm/jimp2/life2/life)
==31913==    by 0x4015F3: main (in /home/sutm/jimp2/life2/life)
==31913==

```

```
==31913== 27,798,664 (126,400 direct, 27,672,264 indirect) bytes in 100 blocks are definitely lost in 1
==31913== at 0x4C2AB80: malloc (in /usr/lib/valgrind/vgpreload_memcheck-amd64-linux.so)
==31913== by 0x4E51F01: ??? (in /lib/x86_64-linux-gnu/libpng12.so.0.50.0)
==31913== by 0x4E4A9B5: png_create_write_struct_2 (in /lib/x86_64-linux-gnu/libpng12.so.0.50.0)
==31913== by 0x4E4AC76: png_create_write_struct (in /lib/x86_64-linux-gnu/libpng12.so.0.50.0)
==31913== by 0x402702: write_png_file (in /home/sutm/jimp2/life2/life)
==31913== by 0x402186: generate (in /home/sutm/jimp2/life2/life)
==31913== by 0x4015F3: main (in /home/sutm/jimp2/life2/life)
==31913==
==31913== LEAK SUMMARY:
==31913== definitely lost: 178,942 bytes in 615 blocks
==31913== indirectly lost: 27,672,264 bytes in 1,399 blocks
==31913== possibly lost: 65,536 bytes in 1 blocks
==31913== still reachable: 279,106 bytes in 16 blocks
==31913== suppressed: 0 bytes in 0 blocks
==31913== Reachable blocks (those to which a pointer was found) are not shown.
==31913== To see them, rerun with: --leak-check=full --show-leak-kinds=all
==31913==
==31913== For counts of detected and suppressed errors, rerun with: -v
==31913== ERROR SUMMARY: 16 errors from 16 contexts (suppressed: 0 from 0)
sutm@jimp:~/jimp2/life2$
```

5 Prezentacja działania aplikacji

1. Przygotowane dane

```
sutm@edi:~/jimp2/life2$ cat data/Acorn
45
30
22 14
23 12
23 14
25 13
26 14
27 14
28 14
sutm@edi:~/jimp2/life2$
```

2. Uruchomienie programu w konsoli

```
sutm@edi:~/jimp2/life2$ ./life data/Acorn data/AcornOutput -n50 -pac -x4
Successfully generated 50 pictures.
sutm@edi:~/jimp2/life2$
```

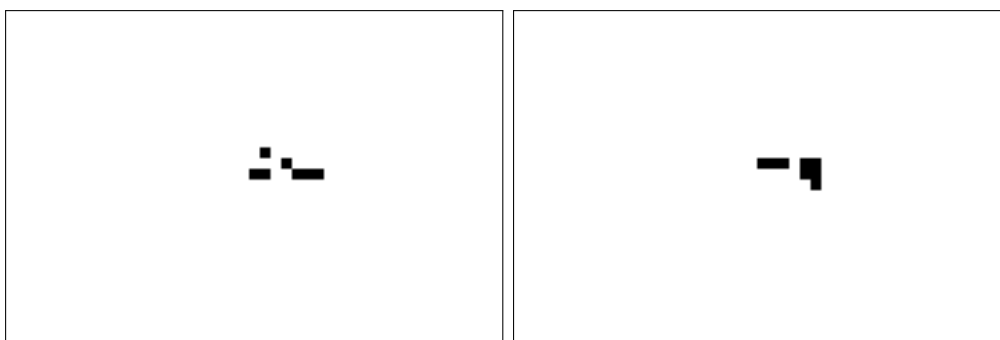
3. Stworzony plik tekstowy z ostatnią generacją










```
sutm@edi:~/jimp2/life2$ ll data
total 60
drwxr-xr-x 2 sutm stud 4096 maj  2 18:57 ./
drwxr-xr-x 4 sutm stud 4096 maj  2 20:39 ../
-rw-r--r-- 1 sutm stud   48 kwi 25 20:18 Acorn
-rw-r--r-- 1 sutm stud  533 maj  2 20:35 AcornOutput
-rw-r--r-- 1 sutm stud  321 kwi 20 17:12 GGun
-rw-r--r-- 1 sutm stud  226 kwi 26 09:04 GgunOut
-rw-r--r-- 1 sutm stud  268 kwi 20 00:34 GliderGun
-rw-r--r-- 1 sutm stud  321 kwi 20 00:36 GliderGuncd
-rw-r--r-- 1 sutm stud  321 maj  2 18:51 GliderGunOut
-rw-r--r-- 1 sutm stud   51 kwi 25 20:34 LWSS
-rw-r--r-- 1 sutm stud   26 maj  2 18:55 LWSSOutput
-rw-r--r-- 1 sutm stud  235 kwi 20 17:11 Pulsar
-rw-r--r-- 1 sutm stud  256 kwi 20 17:13 Pulsar2
-rw-r--r-- 1 sutm stud   36 kwi 20 17:16 Rpentomino
-rw-r--r-- 1 sutm stud  159 kwi 20 17:17 Rpentomino200
sutm@edi:~/jimp2/life2$
```

4. Wygenerowane pliki graficzne w katalogu *pictures/AcornOutput*

```
sutm@edi:~/jimp2/life2$ ll pictures/AcornOutput
total 212
drwxr-xr-x  2 sutm stud 4096 maj  2 20:39 ./
drwxr-xr-x 12 sutm stud 4096 maj  2 18:34 ../
-rw-r--r--  1 sutm stud  251 maj  2 20:35 ac0000.png
-rw-r--r--  1 sutm stud  246 maj  2 20:35 ac0001.png
-rw-r--r--  1 sutm stud  257 maj  2 20:35 ac0002.png
-rw-r--r--  1 sutm stud  266 maj  2 20:35 ac0003.png
-rw-r--r--  1 sutm stud  265 maj  2 20:35 ac0004.png
-rw-r--r--  1 sutm stud  265 maj  2 20:35 ac0005.png
      :
-rw-r--r--  1 sutm stud  386 maj  2 20:35 ac0044.png
-rw-r--r--  1 sutm stud  387 maj  2 20:35 ac0045.png
-rw-r--r--  1 sutm stud  398 maj  2 20:35 ac0046.png
-rw-r--r--  1 sutm stud  407 maj  2 20:35 ac0047.png
-rw-r--r--  1 sutm stud  396 maj  2 20:35 ac0048.png
-rw-r--r--  1 sutm stud  413 maj  2 20:35 ac0049.png
-rw-r--r--  1 sutm stud  415 maj  2 20:35 ac0050.png
sutm@edi:~/jimp2/life2$
```

5. Wygenerowane obrazy



6. Zawartość tekstowego pliku wyjściowego

```
sutm@edi:~/jimp2/life2$ cat data/AcornOutput
45
30
7 14
8 12
8 13
8 14
8 15
9 11
9 12
:
34 11
34 13
34 14
35 5
35 6
35 7
35 10
35 11
35 12
36 5
sutm@edi:~/jimp2/life2$
```

6 Podsumowanie

- Program poprawnie tworzy obrazy z przygotowanych danych wejściowych oraz zapisuje ostatnią generację do pliku tekstowego w takim samym formacie jak dane wejściowe. Dzięki temu, może być on wykorzystany do kontynuowania generacji danej struktury.
- Program odpowiednio obsługuje błędy. Jest odporny na nieprawidłowe dane wejściowe oraz błędne opcje uruchomienia.
- Program pomyślnie przeszedł testy całościowe oraz testy poszczególnych modułów. Test wycieków pamięci zakończył się niepowodzeniem.

Poniżej podsumowanie testów w postaci tabeli:

Test	Wynik
Ogólne działanie programu	Zaliczony
coordinates	Zaliczony
board	Zaliczony
generation	Zaliczony
makepng	Zaliczony
Wycieki pamięci	Niezaliczony