

# MINI\_PAINT

## ОБЯЗАТЕЛЬНАЯ ЧАСТЬ

Напишите программу, которая будет считывать "операционный файл" и печатать результат в консоли

Ваша программа должна принять один аргумент, это будет путь к файлу "operation.it"

Если вашей программе передано больше одного аргумента, напишите "Error: a lot of arguments", а затем `\n` в `STDOUT`

Если вашей программе задано не существующее имя, напишите "Error: name file", а затем `\n` в `STDOUT`

Если какая-либо проблема возникает при открытии и / или чтении "операционного файла", напишите "Error: Operation file corrupted", а затем `\n` в `STDOUT`

Если "операционный файл" не имеет расширение \*.it напишите "Error: Operation file has not correct extension", а затем `\n` в `STDOUT`

"Операционный файл" будет содержать строки с одной операцией в строке

Если строка неверна, то возникает ошибка.  
Последняя строка может быть с или без `\n`

Строки должны быть прочитаны по порядку и поэтому операции должны выполняться в том же порядке

Между каждой переменной в строке должно быть по одному пробелу

## WIDTH HEIGHT BACKGROUND\_CHAR

Эта строка всегда является первой строкой файла и определяет зону, в которой нужно рисовать. Ваша программа не должна отображать ничего за пределами зоны.

- `WIDTH`: должен быть `int` с  $0 < WIDTH \leq 300$ , горизонтальное количество символов, используемых для зоны рисования
- `HEIGHT`: должен быть `int` с  $0 < HEIGHT \leq 300$ , вертикальное количество символов, используемых для зоны рисования
- `BACKGROUND_CHAR`: любое пустое пространство будет заполнено `BACKGROUND_CHAR`

## r X Y WIDTH HEIGHT CHAR

Эта операция нарисует пустой прямоугольник, где будет нарисована только граница прямоугольника

- `r`: символ `r`
- `X`: любое число, горизонтальное положение верхнего левого угла прямоугольника
- `Y`: любое число, вертикальное положение верхнего левого угла прямоугольника
- `WIDTH`: положительное число, но не 0, ширина прямоугольника (горизонтальная)
- `HEIGHT`: положительное число, но не 0, высота прямоугольника (вертикальная)
- `CHAR`: `char` используется для рисования прямоугольника

## R X Y WIDTH HEIGHT CHAR

Эта операция нарисует заполненный прямоугольник

- R: символ R
- X: любое число, горизонтальное положение верхнего левого угла прямоугольника
- Y: любое число, вертикальное положение верхнего левого угла прямоугольника
- WIDTH: положительное число, но не 0, ширина прямоугольника (горизонтальная)
- HEIGHT: положительное число, но не 0, высота прямоугольника (вертикальная)
- CHAR: char используется для рисования прямоугольника

## c X Y RADIUS CHAR

Эта операция нарисует пустой круг, где будет нарисована только граница круга

- c: символ c
- X: float, горизонтальное положение центра круга
- Y: float, вертикальное положение центра круга
- RADIUS: положительный float, но не 0, радиус окружности
- CHAR: символ используется для рисования круга

## C X Y RADIUS CHAR

Эта операция нарисует заполненный круг

- C: символ C
- X: float, горизонтальное положение центра круга
- Y: float, вертикальное положение центра круга
- RADIUS: положительный float, но не 0, радиус окружности
- CHAR: символ используется для рисования круга

## L X Y X1 Y1 CHAR

Эта операция нарисует линию

- L: символ L
- X: float, горизонтальное положение первой точки линии
- Y: float, вертикальное положение первой точки линии
- X1: float, горизонтальное положение второй точки линии
- Y1: float, вертикальное положение второй точки линии
- CHAR: символ используется для рисования линии

Зона рисования разделена на прямоугольники, которые могут содержать по одному символу каждый, мы будем называть их пиксельными

Чтобы сделать все проще, мы будем использовать только верхний левый угол пикселя, чтобы узнать, находится ли этот пиксель в круге или нет

Если расстояние между верхним левым углом пикселя и центром окружности меньше или равно радиусу окружности, то пиксель является частью окружности  
но также:

Пиксель с верхним левым углом на расстоянии больше или равном 1 от границы окружности не является частью пустого круга

Пиксель с верхним левым углом и расстоянием ниже 1 от границы окружности является частью пустого круга.

Примеры:

```
./mini_paint operation
```

Error: Operation file has not correct extension

```
./mini_paint
```

Error: name file

```
./mini_paint operation.it
```

```
10 5 &
```

```
R 1 1 6 3 *
```

```
& & & & & & & & &
& * * * * * & & &
& * * * * * & & &
& * * * * * & & &
& & & & & & & & &
```

Подсказка:

Если у вас есть 2 точки, то они определяются как (Xa, Ya) и (Xb, Yb)

Вы можете получить расстояние между двумя точками по следующей формуле:

$\text{sqrt}((Xa - Xb) * (Xa - Xb) + (Ya - Yb) * (Ya - Yb))$

Бонусная часть будет оцениваться только в том случае, если ваша обязательная часть ИДЕАЛЬНА. Под ИДЕАЛЬНЫМ мы, естественно, подразумеваем, что она должна быть полностью выполнена, что она не может потерпеть неудачу, даже в случаях непредвиденных ошибок. В основном это означает, что если ваша обязательная часть не наберет ВСЕ баллы во время оценки, ваши бонусы будут полностью ПРОИГНОРИРОВАНЫ.

## БОНУСНАЯ ЧАСТЬ

- Если в программу передан второй аргумент "--save", создается файл с изображением и сохраняется с именем result\_operation\_file.it
- Ошибки выводятся красным цветом.
- В строки с фигурами последними символами указывается символы RGB\_R, RGB\_G, RGB\_B, указывающие в какой цвет должны быть покрашены нарисованные фигуры.
  - RGB\_R – красный
  - RGB\_G – зеленый
  - RGB\_B – синий
- Фигуры не могут быть нарисованы тем же символом, что и фоновый символ, иначе предложить пользователю изменить символ в консольном вводе.

## МЕГАБОНУСНАЯ ЧАСТЬ

Напишите программу `mega_bonus_mini_paint`, которая будет считывать файл `result_operation_file.it` и печатать результат в консоли

Ваша программа должна принять один аргумент, это будет путь к файлу `"result_operation_file.it "`

Правила обработки ошибок работы с файлом сохраняются как в обязательной части.

Ваша программа должна проанализировать файл `result_operation_file.it` и вывести в консольный вывод название фигуры изображенной в файле.  
(RECTANGLE, FILL\_RECTANGLE, LINE, CIRCLE, FILL\_CIRCLE)

ПРИМЕР:

Файл `result_operation_file.it`:

```
& & & & & & & & &
& * * * * * & & &
& * * * * * & & &
& * * * * * & & &
& & & & & & & & &
```

Вывод программы:

FILL\_RECTANGLE

## ОЦЕНКА:

Описание	Задание	Балл
Чтение файла	Проверка избыточного количества аргументов	1
	Проверка имени файла	2
	Проверка открытия или чтения файла	1
	Проверка расширения файла	2
	Проверка на отсутствие валидной информации в файле	1
	Проверка на валидность строк	2
Изображение	Проверка на правильность указания ширины и высоты экранной области	3
	Проверка корректности значений X, Y	2
Рисование фигур	Рисование линии	8
	Рисование прямоугольника	4
	Рисование закрашенного прямоугольника	3
	Рисование круга	5
	Рисование закрашенного круга	7