

Корепанова Полина Денисовна

БПИ 207

Архитектура ВС с динамической типизацией

Для тестирования программы использовалась система флагов (-f для чтения из файла и -n для генерации случайных параметров), где входные данные обязаны иметь вид:

*<-f> <Имя файла для чтения > <Имя файла для вывода изначального контейнера> <Имя файла для вывода отсортированного контейнера>*

Или

*<-n> <Количество транспортных средств> <Имя файла для вывода изначального контейнера> <Имя файла для вывода отсортированного контейнера>*

Названия тест файлов:

Для **n**-файла:

*Входные данные:* test/input/test\_input\_**n**.txt

*Выходные данные:* test/output/test\_output\_**n**/out1.txt

test/output/test\_output\_**n**/out2.txt

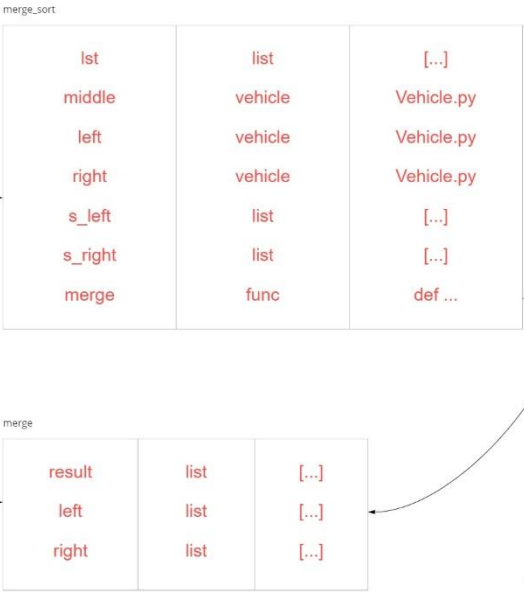
Пример (тест 1): -f test/input/test\_input\_1.txt test/output/out\_test\_1/out1.txt test/output/out\_test\_1/out2.txt

Примечание: для генерации случайных значений были установлены границы, соответствующие среднестатистическим показателям наземного транспорта.

Ниже приведена обобщенная схема разработанной программы.

Схему также можно посмотреть более высокого качества [тут](#).

Память программы	Таблица имен	Память данных	
main.py	container	Container	Container.py
	outputFileName	string	" "
	outputFileName2	string	" "
	file	file	FileName
	capacity	int	<number>
	consumption	float	<number>
	speed	int	<number>
	volume	int	<number>
Container.py	seats	int	<number>
	__init__	func	def ...
	__iter__	func	def...
	__next__	func	def...
	append	func	def...
	size	func	def...
	sort	func	def...
	merge_sort	func	def...
Vehicle.py	merge	func	def...
	data	list	[...]
	index	int	<number>
	__init__	func	def ...
	__lt__	func	def ...
	__le__	func	def ...
	max_distance	func	def ...
	to_string	func	def ...
Car.py	capacity	int	<number>
	consumption	float	<number>
	speed	int	<number>
	__init__	func	def ...
	max_distance	float	<number>
	to_string	func	def ...
Truck.py	capacity	int	<number>
	consumption	float	<number>
	volume	int	<number>
	__init__	func	def ...
	max_distance	float	<number>
Bus.py	to_string	func	def ...
	capacity	int	<number>
	consumption	float	<number>
	seats	int	<number>
	__init__	func	def ...



miro