

ДЗ №2.
Ким Макар БПИ201
Вариант 13

Полученное задание.

Необходимо реализовать программу с обобщённым артефактом в виде растения.

Базовые альтернативы и их уникальные параметры:

- Дерево
 - Возраст – длинное целое
- Кустарник
 - Месяц цветения – перечислимый тип
- Цветок
 - Тип (домашние, садовые, дикие) – перечислимый тип

Общая переменная для всех альтернатив: название – строка символов.

Общая функция: частное от деления числа гласных букв в названии на общую длину названия – действительное число.

Структурная схема архитектуры ВС:

Таблица типов	
int	4
long int	8
double	8
char	1
enum	4
class Random { int first; int last; }	8 4[0] 4[4]
class Plant { char *name[20]; }	160 160[0]
class Tree : Plant { char *name[20]; long int age; }	168 160[0] 8[160]
class Shrub : Plant { char *name[20]; enum flowering_month; }	164 160[0] 4[160]
class Flower : Plant { char *name[20]; enum flow_type; }	164 160[0] 4[160]
class Container { int len; enum max_len; Plant *cont[max_len]; }	80004 4[0] 80000[4]

Память программы		
main(int argc, char *argv[]): int argc char *argv[] Container c FILE *file int size FILE *out1 FILE *out2	80668 4[0] 8[4] 80004[12] 216[80016] 4[80232] 216[80236] 216[80452]	stack stack; heap stack stack; heap stack stack; heap stack; heap
void Out(FILE *file): FILE *file int i	12 8[0] 4[8]	stack; heap stack
void Sort(): int i Plant *curr_val int idx	16 4[0] 8[4] 4[12]	stack stack; heap stack

Характеристики программы:

- Число заголовочных файлов: 6
- Число модулей реализации: 6
- Размер исполняемого файла: 24.4 Кб
- Общий размер исходных текстов: 20 Кб
- Время работы на тестах:

Start	Test1
Stop	
real	0m0.067s
user	0m0.000s
sys	0m0.016s
Start	Test2
Stop	
real	0m0.037s
user	0m0.000s
sys	0m0.000s
Start	Test3
Stop	
real	0m0.038s
user	0m0.000s
sys	0m0.000s
Start	Test4
Stop	
real	0m0.037s
user	0m0.000s
sys	0m0.000s
Start	Test5
Stop	
real	0m0.037s
user	0m0.000s
sys	0m0.000s
Start	Test6
Stop	
real	0m6.336s
user	0m6.297s
sys	0m0.000s

Сравнение с предыдущими реализациями:

По сравнению с функциональным подходом, исполняемый файл весит меньше, а время исполнения на случайных данных (10000) стало меньше на более чем на 20% (7.942s -> 6.336s).