

Постановка задания

Условие: требуется реализовать консольную утилиту архивации/разархивации файлов использующую код Хаффмана [1, 4, 5]. Утилита, назовем ее arj, должна предоставлять следующий интерфейс командной строки

```
arj [-c|-d] -i <filename> -o <filename>
```

- -c и -d - режимы работы утилиты, обозначающие архивацию и разархивацию соответственно;
- опция -i <filename> - указывает файл для архивации/разархивации в зависимости от указанного режима работы;
- опция -o <filename> - указывает файл, в который необходимо сохранить результат работы. Если файл не существует его необходимо создать, если же файл существует необходимо его заменить;

Примечания:

- в качестве входного файла (опция -i) всегда передается существующий файл;
- все каталоги в имени выходного файла существуют до вызова программы;
- опции -c (или -d), -i и -o могут идти в любом порядке;
- в качестве входного файла могут использоваться как текстовые так и бинарные файлы;

Примеры:

```
./arj -c -i ../films/potter.avi -o ../arj/potter.avi.ar
```

```
./arj -d -o /tmp/potter.avi -i ../arj/potter.avi.ar
```

```
./arj -o some_file.ar -i some_file -c
```

```
./arj -i war_and_peace.txt -c -o war_and_peace.txt.ar
```

Правила

Формат сдачи: сдача задания осуществляется через SVN [2, 3] репозиторий <http://mit.spbau.ru/svn/csp13>. Для каждого студента будет создан каталог с именем в формате second_first, где second - фамилия латиницей, а first - имя латиницей. Если вы не нашли своего каталога, или возникли проблемы с SVN сразу же сообщайте.

В вашем каталоге необходимо создать папку ha1, в которую должны попасть исходные коды программы и Makefile. Убедительная просьба не сохранять в каталоге ничего лишнего, например, бинарные файлы или readme.txt или файлы проектов IDE.

Makefile должен быть составлен так, чтобы в результате вызова команды make в каталоге с Makefile появился исполняемый файл с именем arj.

Допустимые компиляторы g++ и clang++ (пользователи Windows могут поставить CygWin или использовать виртуальную машину). Настоятельно рекомендуется использовать флаг -Wall и не допускать предупреждений компилятора.

Чтобы ваша работа была проверена необходимо создать задачу/ticket в трекере и указать в поле владелец/owner фамилию вашего преподавателя практики - oreukin, krinkin или kudinkin соответственно.

Замечания по результатам проверки будут возвращены в виде задачи назначенной на вас. После исправления замечаний нужно вернуть задачу преподавателю и так далее, пока у преподавателя не останется замечаний или не выйдет срок.

Сроки выполнения: последний срок сдачи домашнего задания 8.11.2013 18:00, все работы не принятые к проверке (без соответствующей задачи в трекере) к этому сроку не засчитываются.

Если работа была принята к проверке в срок, то до 15.11.2013 18:00 принимаются исправления к заданию по результатам проверки. Исправления сделанные после 15.11.2013 18:00 не рассматриваются.

Для тех, кто не сдаст. Не получившим зачет по домашнему заданию потребуется выполнить дополнительное задание для допуска к экзамену. Детали будут определены ближе к экзамену.

Список литературы

- [1] Сжатие по алгоритму Хаффмана. <http://algotlist.manual.ru/compress/standard/huffman.php>.
- [2] Системы управления версиями на примере subversion. <http://www.devexp.ru/2009/11/sistema-upravleniya-versiyami-na-primere-subversion-vvedenie/>, 2009.
- [3] Basic svn commands. <http://www.linuxfromscratch.org/blfs/edguide/chapter03.html>, 2012.
- [4] U. V. Vazirani S. Dasgupta, C. H. Papdimitiou. Algorithms. <http://www.cs.berkeley.edu/~vazirani/algorithms/all.pdf>, July 2006. p. 153-155.
- [5] Alexander Simakov. Код Хаффмана. <http://entropyware.info/HuffmanCode/huffcode.html>, Октябрь 2002.