1. (15 points)

This program randomly combines words contained in its input stream and sends them to the output stream. At program execution, the application shall read in the entire set of words to use during the combination process. After reaching the EOF symbol, the program shall then begin randomly selecting words and combining them into a concatenated output word. By default, the program produces only one combined output word for each execution, but users may change this behavior by providing an optional command line argument (through \*argv).

The words the program selects must meet minimum length criteria, and users may also change this value at program execution to meet their needs. The program must accept this.

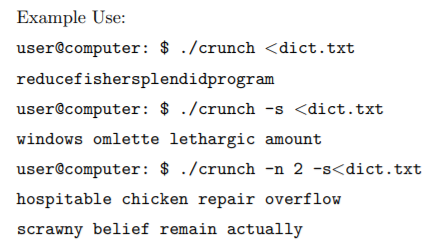
By default, the program squeezes the words together and omits any spacing between them (e.g., somewhatlikethisbutrandom), but users may override this behavior with a command line argument as well, and the program will produce output with the user specified parameter.

The command line arguments may appear in any order.

When unable to construct strings meeting the minimum length from the input stream, the program shall alert the user to an error and suggest corrective action.

There shall be no duplicate words in any single output combination, but the same word may appear twice in the total output when producing more than one string (i.e. -n specified >1)

|  |  |
| --- | --- |
| Program Name: | crunch |
| Command Line Args: | -d <degree> Number of words from input stream to combine in each crunched output (default =4).  -m <size> The minimum length of each word used (default =6).  -n <count> Number of output strings to produce (default =1).  -s Indicates to insert a space between the output words (default=none). |
| Program Input: | stdin |
| Program Output: | stdout |
| Return: | 0 on completion |
| Allowed Imports: | stdio.h stdlib.h time.h |



1. (10 points)

Students shall create a program in C which reads ASCII strings from the standard input until it reaches the EOF symbol. From the accumulated input, the program shall extract only its unique tokens, seperated by the space character, and send these to the standard output.

For example, if provided with a file containing a text document, it shall send to the standard output every unique word contained therein. The order the words appear in the output does not need to be sorted in any particular order.

The program shall insert a space character between each output word and ignore line endings.

|  |  |
| --- | --- |
| Program Name: | unique |
| Command Line Args: | none |
| Program Input: | stdin |
| Program Output: | stdout |
| Return: | 0 on completion |
| Allowed Imports: | Stdio.h |

Example Use:

user@computer: $ ./unique<iamsam.txt

1. (5 points)

Use the grammar defined below to construct a parse tree for the following statement:

1. (5 points)

Given the following simple grammar, indicate if the following sentences are valid with a T for true or F for false.

(a) (1 point) .... baab

(b) (1 point) .... bbbab

(c) (1 point) .... bbaaaaa

(d) (1 point) .... bbaab

(e) (1 point) .... baabbb

1. (5 points)

Given the following simple grammar, indicate if the following sentences are valid with a T for true or F for false.

1. (1 point) 1X,2Y,3Z
2. (1 point) 4Z,3Y,2X,1A
3. (1 point) X2,2X,Y3,1Z
4. (1 point) 3Y,4X,ZZ,1X
5. (1 point) 1Z,4Y,XZ,3X