```
Yang Zheng
CS347 Spring23
Assignment1
```

```
/* CSCI347 Spring23
 * Assignment 1
 * Modified April 3, 2023 Yang zheng
 */

#include <stdio.h>
#include <string.h>
#include <unistd.h>
#include <errno.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/wait.h>
#include <stdbool.h>

/* Constants */

#define LINELEN 1024

/* Prototypes */

void processline (char *line);

void off_quote(char *line) {
  int j = 0;
  int lineLength = strlen(line);
  for (int i = 0; i < lineLength; i++) {
    if (line[i] != '"') {
        line[j++] = line[i];
    }
  }
  line[j] = '\0';
}

char** arg_parse (char *line, int *argcptr) {
  int count = 1;
  int i = 0;
  bool no_quote = true;
  int length = strlen(line);

  while (line[i] != 0 && i < length) {
    if (line[i] != ' ') {
      while (line[i] != 0 && i < length) {
              if (line[i] == '"') {
        no_quote = !no_quote;
       }
       if (line[i] == ' ') {
         if (no_quote == false) { // if we have read a \", don't do anything
           ;
         } else {
           count++;
           break;
         }
       }
       i++;
      }
      i++;
    } else {
      i++;
    }
```

```
  }

  if (no_quote == false) {
    fprintf(stderr, "No matching double quotes");
  }

  i = 0;
  int j = 0;

  char** arr = (char**) malloc ((count + 1) * sizeof(char*));
  if (arr == NULL) {
     fprintf (stderr, "Failed to malloc");
  }

  while (line[i] != 0 && i < length) {
    if (line[i] != ' ') {
      arr[j] = &line[i];
      j++;
      while (line[i] != 0 && i < length) {
              if (line[i] == '"') {
          no_quote = !no_quote;
        }
        if (line[i] == ' ') {
          if (no_quote == false) { // if we have read a \", don't do anything
            ;
          } else {
            line[i] = 0;
            break;
          }
        }
        i++;
      }
      i++;
    } else {
      i++;
    }
  }

  for (int i = 0; i < j; i++) {
    off_quote(arr[i]);
  }


  arr[count] = NULL;
  *argcptr = count;
  return arr;
}


/* Shell main */
int
main (void)
{
    char   buffer [LINELEN];    int    len;

    while (1) {

        /* prompt and get line */
        fprintf (stderr, "%% ");
```

```
        if (fgets (buffer, LINELEN, stdin) != buffer)
          break;

        /* Get rid of \n at end of buffer. */
        len = strlen(buffer);
        if (buffer[len-1] == '\n')
            buffer[len-1] = 0;

        /* Run it ... */
        processline (buffer);

    }

    if (!feof(stdin))
        perror ("read");

    return 0;           /* Also known as exit (0); */
}


void processline (char *line)
{
    pid_t  cpid;
    int    status;

    int argc;
    char** p_arr = arg_parse(line, &argc);
    if (line == NULL) {
      return;
    }

    /* Start a new process to do the job. */
    cpid = fork();
    if (cpid < 0) {
      /* Fork wasn't successful */
      perror ("fork");
      return;
    }

    /* Check for who we are! */
    if (cpid == 0) {
      /* We are the child! */
      execvp(p_arr[0], p_arr);
      /* execlp reurned, wasn't successful */
      perror ("exec");
      fclose(stdin);  // avoid a linux stdio bug
      exit (127);
    }

    /* free pointer array */
    p_arr = NULL;
    free(p_arr);

    /* Have the parent wait for child to complete */
    if (wait (&status) < 0) {
      /* Wait wasn't successful */
      perror ("wait");
    }
```

```
}
```

```
Script started on 2023-04-05 00:09:45-07:00 [TERM="xterm-256color" TTY="/dev/pts
/2" COLUMNS="80" LINES="24"]
^[[?2004h^[]0;zhengy@cf405-10: ~/csci347/csci347_s23/ush^G^[[01;32mzhengy@cf405-
10^[[00m:^[[01;34m~/csci347/csci347_s23/ush^[[00m$ gcc ush.c
^[[?2004l
^[[?2004h^[]0;zhengy@cf405-10: ~/csci347/csci347_s23/ush^G^[[01;32mzhengy@cf405-
10^[[00m:^[[01;34m~/csci347/csci347_s23/ush^[[00m$ ./a.out
^[[?2004l
% ./printArg 1 2 3
arg0 is: ./printArg
arg1 is: 1
arg2 is: 2
arg3 is: 3
% ^[[?2004h^[]0;zhengy@cf405-10: ~/csci347/csci347_s23/ush^G^[[01;32mzhengy@cf40
5-10^[[00m:^[[01;34m~/csci347/csci347_s23/ush^[[00m$ exit
^[[?2004l
exit

Script done on 2023-04-05 00:10:06-07:00 [COMMAND_EXIT_CODE="0"]
```

```
Script started on 2023-04-05 11:16:33-07:00 [TERM="xterm-256color" TTY="/dev/pts
/0" COLUMNS="80" LINES="24"]
^[[?2004h^[]0;zhengy@cf405-07: ~/csci347/csci347_s23/ush^G^[[01;32mzhengy@cf405-
07^[[00m:^[[01;34m~/csci347/csci347_s23/ush^[[00m$ gcc ush^H^[[K^H^[[K^H^[[Kush.
c
^[[?2004l
^[[?2004h^[]0;zhengy@cf405-07: ~/csci347/csci347_s23/ush^G^[[01;32mzhengy@cf405-
07^[[00m:^[[01;34m~/csci347/csci347_s23/ush^[[00m$ ./a.out
^[[?2004l
% prog "this is a single arg"
exec: No such file or directory
% ls
 a1.pdf        core.672417   core.684825   printArg      test
 a1.ps         core.672520   core.684994   printArg.c    test.c
 a1.ps~        core.672620   core.685049   printArg.c~   ush
 a.out         core.672887   core.831541   report.pdf   '#ush.c#~'
 core.668271   core.673431   fully         report.ps     ush.c
 core.671057   core.684774   header.txt    script-nq
% ./arg^H ^H^H ^H^H ^HprintArg "hell"o World
arg0 is: ./printArg
arg1 is: hello
arg2 is: World
% ^[[?2004h^[]0;zhengy@cf405-07: ~/csci347/csci347_s23/ush^G^[[01;32mzhengy@cf40
5-07^[[00m:^[[01;34m~/csci347/csci347_s23/ush^[[00m$ exit
^[[?2004l
exit

Script done on 2023-04-05 11:17:24-07:00 [COMMAND_EXIT_CODE="0"]
```