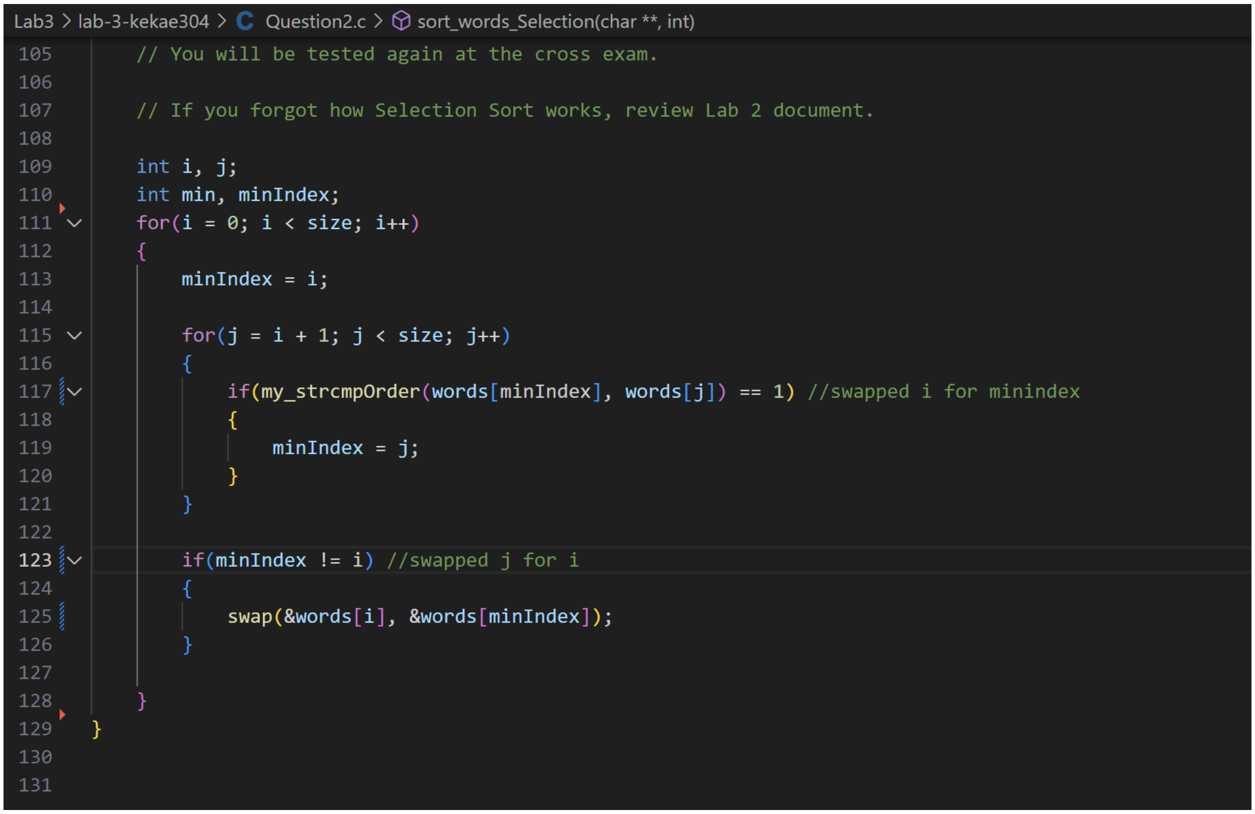

Lab3 > lab-3-kekae304 > Question2.c > sort_words_Selection(char **, int)

```c
    // You will be tested again at the cross exam.

    // If you forgot how Selection Sort works, review Lab 2 document.

    int i, j;
    int min, minIndex;
    for(i = 0; i < size; i++)
    {
        minIndex = i;

        for(j = i + 1; j < size; j++)
        {
            if(my_strcmpOrder(words[minIndex], words[j]) == 1) //swapped i for minindex
            {
                minIndex = j;
            }
        }

        if(minIndex != i) //swapped j for i
        {
            swap(&words[i], &words[minIndex]);
        }

    }
}


```

The changes have been made here and are annotated in comments!

GDB:

```
(gdb) run
Starting program: C:\Users\erion\OneDrive\Documents\Comp Eng
2SH4\Lab3\lab-3-kekae304/.\Lab3.exe
[New Thread 20224.0x1858]
[New Thread 20224.0x2268]

Breakpoint 1, sort_words_Selection (
    words=0xd611f8, size=6)
    at Question2.c:116
116         for(i = 0; i < size; i++)
(gdb) i locals
i = 6
j = 2004719200
minIndex = 14029304
(gdb) step
118             minIndex = i;
(gdb) i locals
i = 0
j = 2004719200
```

```
minIndex = 14029304
(gdb) step
120             for(j = i + 1; j < size; j++)
(gdb) i locals
i = 0
j = 2004719200
minIndex = 0
(gdb) step
122                 if(my_strcmpOrder(words[minIndex], words[j]) == 1) //swapped i for minindex
(gdb) i locals
i = 0
j = 1
minIndex = 0
(gdb) step
my_strcmpOrder (str1=0xd61228 "milan",
    str2=0xd65000 "hello")
    at Question1.c:73
73              const char *p1 = str1; //creating a pointer to the first item of string 1
(gdb) i locals
p1 = 0xfffffffe <Address 0xfffffffe out of bounds>
p2 = 0xe9bb833f <Address 0xe9bb833f out of bounds>
(gdb) step
74              const char *p2 = str2; //creating a pointer to the first item of string 2
(gdb) i locals
p1 = 0xd61228 "milan"
p2 = 0xe9bb833f <Address 0xe9bb833f out of bounds>
(gdb) step
76              while (*p1 && *p2) //while loop -> while both of the pointers are pointing to valid
items
(gdb) i locals
p1 = 0xd61228 "milan"
p2 = 0xd65000 "hello"
(gdb) step
78                      if (*p1 < *p2) //checking to see if pointer of string 1 is smaller than pointer of
string 2
(gdb) i locals
p1 = 0xd61228 "milan"
p2 = 0xd65000 "hello"
(gdb) step
82                      else if (*p1 > *p2) // checking to see if pointer of string 1 is now bigger than
pointer of string 2
(gdb) i locals
p1 = 0xd61228 "milan"
p2 = 0xd65000 "hello"
```

```
(gdb) step
84                          return 1; // returning a 1 because string 1 lexicographicsally bigger than
the string 2
(gdb) i locals
p1 = 0xd61228 "milan"
p2 = 0xd65000 "hello"
(gdb) step
102     }
(gdb) i locals
p1 = 0xd61228 "milan"
p2 = 0xd65000 "hello"
(gdb) step
sort_words_Selection (words=0xd611f8,
   size=6) at Question2.c:124
124                   minIndex = j;
(gdb) i locals
i = 0
j = 1
minIndex = 0
(gdb) step
120             for(j = i + 1; j < size; j++)
(gdb) i locals
i = 0
j = 1
minIndex = 1
(gdb) step
122                 if(my_strcmpOrder(words[minIndex], words[j]) == 1) //swapped i for minindex
(gdb) i locals
i = 0
j = 2
minIndex = 1
(gdb) step
my_strcmpOrder (str1=0xd65000 "hello",
   str2=0xd65118 "programming")
   at Question1.c:73
73             const char *p1 = str1; //creating a pointer to the first item of string 1
(gdb) i locals
p1 = 0xd61228 "milan"
p2 = 0xd65000 "hello"
(gdb) step
74             const char *p2 = str2; //creating a pointer to the first item of string 2
(gdb) i locals
p1 = 0xd65000 "hello"
p2 = 0xd65000 "hello"
```

```
(gdb) step
76              while (*p1 && *p2) //while loop -> while both of the pointers are pointing to valid
items
(gdb) i locals
p1 = 0xd65000 "hello"
p2 = 0xd65118 "programming"
(gdb) step
78                      if (*p1 < *p2) //checking to see if pointer of string 1 is smaller than pointer of
string 2
(gdb) i locals
p1 = 0xd65000 "hello"
p2 = 0xd65118 "programming"
(gdb) step
80                              return 0; //returning 0 because string 1 is lexicographically smaller than
string 2
(gdb) i locals
p1 = 0xd65000 "hello"
p2 = 0xd65118 "programming"
(gdb) step
102     }
(gdb) i locals
p1 = 0xd65000 "hello"
p2 = 0xd65118 "programming"
(gdb) step
sort_words_Selection (words=0xd611f8,
    size=6) at Question2.c:120
120             for(j = i + 1; j < size; j++)
(gdb) i locals
i = 0
j = 2
minIndex = 1
(gdb) step
122                 if(my_strcmpOrder(words[minIndex], words[j]) == 1) //swapped i for minindex
(gdb) i locals
i = 0
j = 3
minIndex = 1
(gdb) step
my_strcmpOrder (str1=0xd65000 "hello",
    str2=0xd65230 "apple")
    at Question1.c:73
73              const char *p1 = str1; //creating a pointer to the first item of string 1
(gdb) step
74              const char *p2 = str2; //creating a pointer to the first item of string 2
```

```
(gdb) step
76              while (*p1 && *p2) //while loop -> while both of the pointers are pointing to valid
items
(gdb) step
78                      if (*p1 < *p2) //checking to see if pointer of string 1 is smaller than pointer of
string 2
(gdb) step
82                      else if (*p1 > *p2) // checking to see if pointer of string 1 is now bigger than
pointer of string 2
(gdb) step
84                              return 1; // returning a 1 because string 1 lexicographicsally bigger than
the string 2
(gdb) step
102     }
(gdb) step
sort_words_Selection (words=0xd611f8,
    size=6) at Question2.c:124
124                     minIndex = j;
(gdb) step
120             for(j = i + 1; j < size; j++)
(gdb) step
122                 if(my_strcmpOrder(words[minIndex], words[j]) == 1) //swapped i for minindex
(gdb) step
my_strcmpOrder (str1=0xd65230 "apple",
    str2=0xd65348 "zebra")
    at Question1.c:73
73              const char *p1 = str1; //creating a pointer to the first item of string 1
(gdb) step
74              const char *p2 = str2; //creating a pointer to the first item of string 2
(gdb) step
76              while (*p1 && *p2) //while loop -> while both of the pointers are pointing to valid
items
(gdb) step
78                      if (*p1 < *p2) //checking to see if pointer of string 1 is smaller than pointer of
string 2
122                 if(my_strcmpOrder(words[minIndex], words[j]) == 1) //swapped i for minindex
(gdb) step
my_strcmpOrder (str1=0xd65230 "apple",
    str2=0xd65460 "banana")
    at Question1.c:73
73              const char *p1 = str1; //creating a pointer to the first item of string 1
(gdb) step
74              const char *p2 = str2; //creating a pointer to the first item of string 2
(gdb) quit
```