4.47

#include <stdio.h>

#include <stdlib.h>

/\*Bubble sort:Pointer version\*/

void bubble\_p(long \*data,long count){

long i,last;

for(last=count-1;last>0;last--){

for(i=0;i<last;i++)

if(\*(data+i+1)<\*(data+i)){

/\*Swap adjacent element\*/

long t=\*(data+i+1);

\*(data+i+1)=\*(data+i);

\*(data+i)=t;

}

}

}

int main()

{

long data[10] = {2,1,5,3,4,10,7,8,6,9};

bubble\_p(data,10);

for(int i=0;i<10;i++){

printf("%ld ",data[i]);

}

return 0;

}

Y86-64:

#Execution begins at address 0

.pos 0

irmovq stack,%rsp #Set up stack pointer\_safety

call main #Excute main program

halt #Terminate program

#Array of 10 Elements

.align 10

array:

.quad 0x000000000001

.quad 0x000000000002

.quad 0x000000000003

.quad 0x000000000004

.quad 0x000000000005

.quad 0x000000000006

.quad 0x000000000007

.quad 0x000000000008

.quad 0x000000000009

.quad 0x00000000000a

main:

irmovq array,%rdi

irmovq $4,%rsi

call bubble\_p

ret

#long bubble\_p(long \*data, long count)

#data in %rdi,count in %rsi

bubble\_p:

irmovq $8,%r8

irmovq $1,%r9

rrmovq %rsi,%rax

jmp test1

loop1:

xorq %rcx,%rcx

loop2:

rrmovq %rcx,%rdx

andq %rdx,%rdx

andq %rdx,%rdx

andq %rdx,%rdx

addq %rdi,%rdx

mrmovq (%rdx),%r10

addq %r8,%rdx

mrmovq (%rdx),%r11

rrmovq %r11,%r12

subq %r10,%r12

jge test2

rmmovq %r10,(%rdx)

subq %r8,%rdx

rmmovq %r11,(%rdx)

test2:

rrmovq %rcx,%rdx

subq %r9,%rdx

jne loop2

test1:

subq %r9,%rax

jne loop1

ret

#Stack starts here and grows to lower addresses

.pos 0x200

stack: