Function parameters max number

I did not find any limitation of count function parameters in the C99 standard and I guess it is only limited by stack size.

However I've written a simple test program to demonstrate the behavior of a function with a large count of parameters. When its about 10k, I get the following error on gcc (gcc version 4.5.3 on Cygwin):

```
/usr/lib/gcc/i686-pc-cygwin/4.5.3/../../../libcygwin.a(libcmain.o):(.text+0xa9):
undefined reference to `_WinMain@16'
```

I realize that such large count of parameters is unlikely but I wonder what parameter of the compiler determines this limit?

EDIT

script to generate C-source

```
#!/bin/sh

num=$1

echo "" > out.c
echo "#include <stdio.h>" >> out.c

echo "int getsum( " >> out.c

i=0
while [ $i -lt $num ]
do
    ((i++))
    if [ $i -eq $num ]
    then
        echo "int p$i )" >> out.c
    else
        echo -ne "int p$i," >> out.c
    fi
done

echo "{" >> out.c

echo -ne "  return " >> out.c

i=0
while [ $i -lt $num ]
do
    ((i++))
        if [ $i -eq $num ]
        then
                echo "p$i;" >> out.c
        else
                echo -ne "p$i + " >> out.c
        fi
done

echo "}" >> out.c

echo "int main(){"  >> out.c
echo "printf(\"Sum of %d elements is %d\", $num, getsum(" >> out.c

i=0
while [ $i -lt $num ]
do
        ((i++))
        if [ $i -eq $num ]
        then
                echo "$i" >> out.c
        else
                echo -ne "$i," >> out.c
        fi
done

echo "));" >> out.c

echo "return 0;}" >> out.c
gcc out.c
./a.exe
```

c parameter-passing stack-overflow

edited Jan 27 '12 at 14:41

asked Jan 27 '12 at 14:30
triclosan
3,474 ● 2 ● 17 ● 38

Care to share the code that caused this? – Martin Green Jan 27 '12 at 14:35

@Martinsh Shaiters i've wrote some sh-script to generate sources. See updt – triclosan Jan 27 '12 at 14:37

1 Answer

The Standard specifies a certain minimum number which every implementation must support,

5.2.4.1 Translation Limits

> — 127 parameters in one function definition
> — 127 arguments in one function call

edited Jan 27 '12 at 16:52
jørgensen
7,464 ● 1 ● 9 ● 19

answered Jan 27 '12 at 14:34
Alok Save
142k ● 25 ● 277 ● 425

Interesting. That's not a whole lot! I would have thought "unlimited, depending on memory size" too. – Mr Lister Jan 27 '12 at 14:56

1 These are the *minimum acceptable limits* an implementation may impose. It's certainly allowable (and desirable) to have a larger limit or no explicit limit at all on things like this. – R.. Jan 27 '12 at 16:33

1 There is also the limit on the maximum length of a source line, 1K-something, or maybe 4K-something. The test program will likely hit this limit, too. (whatever comes first ...) – wildplasser Jan 27 '12 at 17:00

9 Not to mention there is a *maximum acceptable limit* imposed by the humans who have to read your code, and exceeding this limit can cause Undefined Behavior in those persons. So don't stretch it :) – jørgensen Jan 27 '12 at 17:43

@wildplasser thank you for this notation – triclosan Jan 28 '12 at 12:11