Summary Exercise - Week 5

Due Nov 3 at 11:59pm **Points** 21 **Questions** 21
Available after Oct 27 at 12am **Time Limit** 360 Minutes **Allowed Attempts** 2

Take the Quiz Again

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	43 minutes	21 out of 21

Score for this attempt: **21** out of 21
Submitted Oct 27 at 6:39pm
This attempt took 43 minutes.

Question 1

1 / 1 pts

The four-byte sequence 0x86 0x65 0x53 0x82 stored in consecutive memory cells in a little-endian architecture represents _____ (decimal) when interpreted as a 32-bit signed integer.

Correct!

-2,108,463,738

orrect Answers

-2,108,463,738 (with margin: 0)

Question 2

1 / 1 pts

The four-byte sequence 0xAB 0xAA 0x2A 0x96 stored in consecutive memory cells in a little-endian architecture represents _____ (decimal) when interpreted as a 32-bit signed integer.

Correct!

-1,775,588,693

orrect Answers

-1,775,588,693 (with margin: 0)

Question 3

1 / 1 pts

The four-byte sequence 0xCB 0x2A 0x2F 0x9A stored in consecutive memory cells in a little-endian architecture represents _____ (decimal) when interpreted as a 32-bit signed integer.

Correct!

-1,708,184,885

orrect Answers

-1,708,184,885 (with margin: 0)

Question 4

1 / 1 pts

The four-byte sequence 0xAA 0x8B 0x8C 0x4B stored in consecutive memory cells in a little-endian architecture represents _____ (decimal) when interpreted as a 32-bit signed integer.

Correct!

1,267,501,994

orrect Answers

1,267,501,994 (with margin: 0)

Question 5

1 / 1 pts

The four-byte sequence 0x8D 0x0C 0x52 0xDB stored in consecutive memory cells in a little-endian architecture represents _____ (decimal) when interpreted as a 32-bit unsigned integer.

Correct!

3,679,587,469

orrect Answers

3,679,587,469 (with margin: 0)

Question 6

1 / 1 pts

The four-byte sequence 0x22 0x88 0x8A 0x56 stored in consecutive memory cells in a little-endian architecture represents _____ (decimal) when interpreted as a 32-bit unsigned integer.

Correct!

1,451,919,394

orrect Answers

1,451,919,394 (with margin: 0)

Question 7

1 / 1 pts

Which offers a more flexible approach, passing arguments to procedures in registers, or on the stack?

Correct!

- (•) on the stack
- ( ) in registers

Question 8

1 / 1 pts

What advantages do stack parameters have over register parameters?

- [ ] Register parameters are optimized for speed.

Correct!

- [x] Stack parameters are compatible with high-level languages.

Correct!

- [x] Stack parameters reduce code clutter because registers do not have to be saved and restored.

- [ ] Programs using stack parameters execute more quickly.

Question 9

1 / 1 pts

Local variables are created by adding a positive value to the stack pointer.

- ( ) True

Correct!

- (•) False

Question 10

1 / 1 pts

Assuming that a procedure contains no local variables, a stack frame is created by which sequence of actions at runtime?

- ( ) arguments pushed on stack; procedure called; EBP set to ESP; EBP pushed on stack

- ( ) EBP pushed on stack; arguments pushed on stack; procedure called; EBP set to ESP

- ( ) arguments pushed on stack; EBP pushed on stack; EBP set to ESP; procedure called

Correct!

arguments pushed on stack; procedure called; EBP pushed on stack; EBP set to ESP

Question 11

1 / 1 pts

When an argument is passed by value, a copy of the address is pushed on the stack.

True

Correct!

False

Question 12

1 / 1 pts

Values passed to a subroutine by a calling program are called _____.

Correct!

arguments

orrect Answers

arguments

8.1

Question 13

1 / 1 pts

A subroutine's stack frame always contains the caller's return address and the subroutine's local variables.

Correct!

- (•) True
- ( ) False

Question 14

1 / 1 pts

A stack frame is _____

- ( ) A register window pointing to local variables.
- ( ) The area of the stack set aside for storing global strings.

Correct!

- (•) The area of the stack set aside for passed arguments, subroutine return address, local variables, and saved registers.
- ( ) An area in the heap that is used to store global variables
- ( ) The area of the text segment set aside for passed arguments, subroutine return address, local variables, and saved registers

Question 15

1 / 1 pts

High-level languages always pass arrays to subroutines by value.

- ( ) True

Correct!

- (•) False

Question 16

1 / 1 pts

An argument passed by reference consists of the offset of an object.

Correct!

- (•) True
- ( ) False

Question 17

1 / 1 pts

Another name for a stack frame is

- ( ) Heap record
- ( ) Vinyl record
- ( ) Aviation record
- ( ) Local storage

Correct!

- (•) Activation record
- ( ) Stack record

Question 18

1 / 1 pts

What are the two common types of stack parameters?

- [ ] Formal parameters.

Correct!

- [x] Value parameters

Correct!

- [x] Reference parameters
- [ ] Abstract parameters.
- [ ] Object parameters.
- [ ] Static parameters.

Question 19

1 / 1 pts

Which action must take place inside a procedure to reserve space on the stack for two doubleword local variables?

- [ ] after MOV EBP,ESP, add 8 to the stack pointer (ESP)
- [ ] after PUSH EBP, add 8 to the base pointer (EBP)
- [ ] after PUSH EBP, subtract 8 from the base pointer (EBP)

Correct!

- [x] after MOV EBP,ESP, subtract 8 from the stack pointer (ESP)

Question 20

1 / 1 pts

Passing by reference requires accessing a parameter's offset on the stack inside the called procedure.

Correct!

- [x] True
- [ ] False

Question 21

1 / 1 pts

What general types of parameters are passed on the stack?

- [ ] Parent-child arguments
- [ ] Evaluation arguments
- [ ] Legal arguments
- [x] Reference arguments — Correct!
- [x] Value arguments — Correct!
- [ ] Context-free arguments

Quiz Score: **21** out of 21