|  |  |
| --- | --- |
| NAME |  |

#### General Instructions

**EVEN Version Solution**

* **CLOSED BOOK**
* **Devices with Internet accesses such as laptops, tablets, smartphones etc. are NOT allowed.**
* If the information provided is not sufficient to answer a question, make reasonable assumptions.
* *Maximum time allotted for this examination is 100 minutes.*
* *Total points = 100*
* **Calculators are allowed if they are not connected to the Internet.**
* **Please write legibly, such that it is easy for the TA’s to grade. You can write your answers electronically in a word file or on a piece of paper.**

*This is a formal EXAMINATION. You are not supposed to help or seek help from any external or internal source, including the World Wide Web, solution manuals etc.*

|  |  |  |
| --- | --- | --- |
| **Question** | **Max Points** | **Your Score** |
| **1** | **10** |  |
| **2** | **10** |  |
| **3** | **20** |  |
| **4** | **15** |  |
| **5** | **15** |  |
| **6** | **10** |  |
| **7** | **10** |  |
| **8** | **5** |  |
| **9** | **5** |  |
| **Total** | **100** |  |

**Use the time wisely. Do not waste time on one question.**

**Read the question carefully.**

**Make sure the answers are clear and precise.**

**Show your work.**

**Use the back of the page if space is not sufficient to answer a question. Indicate clearly that your answer is on the back of the page.**

**Question 1 (10 points)**

If we increase the voltage of the processor by double, from 2V to 3V how much power increase in processor are we expecting to have? Power=CV2F

34/24 = 81/16 = 5.06

If they did 33/23 or 32/22 if they explain their assumption, they should also get a full grade. But if they do not explain their assumption, they only get half of the grade.

half points for partial solution.

**Question 2 (10 points)**

Assume that floating-point instructions can be made 5 times faster. What fraction of the time in the original program should an application spend on floating point code, so that the speed up of the overall application is 3?

E\_old = 100 = x + (100 – x) where is x is the time spent on FP and 100-x is rest

E\_new = x/5 + (100 – x) because FP can be made 5 times faster

E\_new = 100/3 because speed up overall is 3

So, x/5 + (100 – x) = 100/3 so x = 1000/12 = 83.33

So, 83.33% of the time should be spent on floating point code to get a 3x speed up through FP acceleration.

half points for partial solution.

**Question 3 (8x2.5=20 points)**

Please provide short answers to the following questions:

(a) Describe 2 ways by which you can improve the yield in the manufacturing of a processor:

1. Reduce die size
2. Reduce defects per unit area

(b) What does compiler do if it runs out of register for a procedure call?

Use stack

(c) Which of power or energy optimization is most important in mobile processors?

energy

(d) How about server scale architectures?

Power

(e)Which one of stack or heap is larger? Or same size?

Heap is larger

(f)Which one of stack or heap (or both) should be deallocated manually after use?

heap

(g) Which one of stack or heap can be accessed faster?

Stack

(h) What is the best metric of performance to compare different computers?

1) Instructions executed per cycle

2) Execution time in seconds

3) Operating frequency

4) Size of cache and memory subsystem

**#2**

**Question 4 (2x7.5=15 points)**

Consider the two-following sequenced of the same code compiled with two different methods:

a) compare the CPI for the two methods

CPI1 = (1x1+2x5+4x2)/ (1+2+4) = 19/7 = 2.7

CPI2 =(4x1+1x5+6x2)/(4+1+6)= 21/11 = 1.9

If IPC was given, 50% awarded, all other mistakes no credit.

b) which one is faster and by how much?

Assuming they are running at the same clock frequency sequence 1 is faster, as it takes fewer cycles to execute. Always execution time is the only metric to compare performance.

21/19 = 1.1 sequence 1 is faster by almost 10% or so

If improvement given in clock cycles points awarded, all other mistakes no credit.

|  |  |  |  |
| --- | --- | --- | --- |
| Class | A | B | C |
| CPI for class | 1 | 5 | 2 |
| IC in compiled sequence 1 | 1 | 2 | 4 |
| IC in compiled sequence 2 | 4 | 1 | 6 |

**Question 5 (15 points)**

**Translate the following piece of code into assembly. Assume that the input (variable a) is stored in x5 and the final answer (sum) must be stored in x10.**

Diagram, schematic

Description automatically generated

Students may have assumed “a” is already in x5. 6 points for first 2 lines. 3 points for proper branch structure (only the condition where “a” is 0 will exit the loop). 3 points for properly incrementing “sum”. 3 points for properly decrementing “a” (note the sub instruction doesn’t have an immediate, if used, it requires 3 register operands).

****

**Question 6 (10 points)**

**What is the content of memory after execution? Consider that arrayOfNums is stored at address 0x10000000.**

A picture containing text

Description automatically generated

**Option A) Option B)**

Table

Description automatically generatedTable

Description automatically generated

Table

Description automatically generated**Option C) Option D)**

Table

Description automatically generated

**Answer: C**

**|**

**Question 7 (10 points)**

Which of the following is assembly translation of the while loop below?

while (save[i] == k) i += 3;

(Assumption: i in x22, k in x24, address of save in x25)

|  |  |
| --- | --- |
| **a)**  **Loop: slli x10, x22, 3**  **add x10, x10, x25**  **ld x9, 0(x10)**  **bne x9, x24, Exit**  **addi x22, x22, 1**  **beq x0, x0, Loop**  **Exit:** | **b)**  **Loop: slli x10, x22, 3**  **add x10, x10, x25**  **ld x9, 0(x10)**  **bne x9, x24, Exit**  **addi x22, x22, 3**  **beq x0, x0, Loop**  **Exit:** |
| **c)**  **Loop: slli x10, x22, 8**  **add x10, x10, x25**  **ld x9, 0(x10)**  **bne x9, x24, Exit**  **addi x22, x22, 1**  **beq x0, x0, Loop**  **Exit:** | **d)**  **Loop: slli x10, x22, 8**  **add x10, x10, x25**  **ld x9, 0(x10)**  **bne x9, x24, Exit**  **addi x22, x22, 3**  **beq x0, x0, Loop**  **Exit:** |

**Answer: b**

**Question 8 (5 points)**

Procedure **a** calls procedure **b**. Complete the following statements:

1. Procedure is responsible for saving any saved registers that will be overwritten on the stack.
2. To return from procedure **b**, the instruction should be used. (name the instruction)
3. Procedure **a** must save the next instruction to execute after a return from **b** in the register .

**Answer: a) b; b) jalr; c) x1 or ra**

1.5 for each correct answer. Full credits only if all 3 are correct.

**Question 9 (5 points)**

Which of the following RISC-V assembly codes could the C code below compile to?

Assume **a**, **b**, and **c** are in registers **x10-x12**, respectively, and the result goes in **x13**.



1. 
2. 
3. 
4. 

**Answer: c**