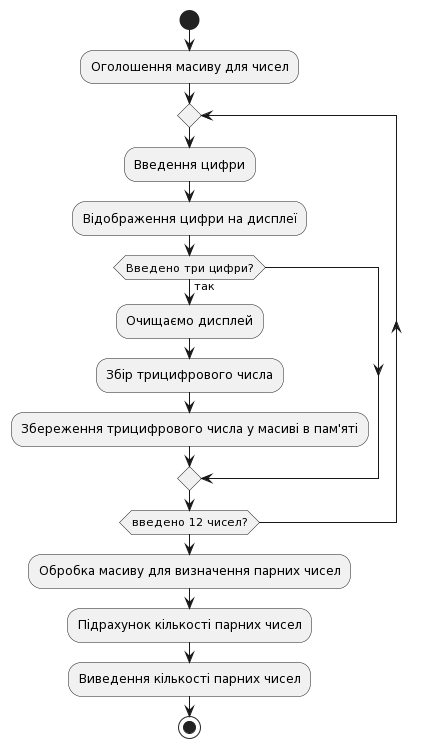
**Звіт. Лабораторна робота №7.**

**«**Робота з масивами та програмування циклу з переадресацією**»**

**Зміст звіту:**

1. Формулювання варіанта завдання та його опис.
2. Граф-схема алгоритму розв'язання задачі.
3. Схема розподілу пам'яті (розміщення в ОЗП змінних, програми та необхідних констант).
4. Записати в таблицю фінальний текст програми із коментарями.
5. Скріншоти працюючої програми.

**Текст звіту**



*Текст програми студента із коментарями.   
При необхідності додайте в таблицю рядки та пронумеруйте їх*

|  |  |  |  |
| --- | --- | --- | --- |
| **Адреса комірки** | **Команда процесора** | | **Опис виконуваної дії** |
| **Мнемокод** | **Код команди** |
| 000 | CALL 148 | 190148 | *Очищаю дисплей* |
| 001 | CALL 155 | 190155 | Очищаю буфер клавіатури |
| 002 | RD #500 | 211500 | Читаю число 500 – комірка пам’яті, де буде почато масив |
| 003 | WR R0 | 320000 | Записую в R0 |
| 004 | RD #000 | 211000 | Читаю 0, буде використовуватись для підрахунку, яке це по порядку буде число |
| 005 | WR R1 | 310001 | Записую в регістр |
| 006 | RD #003 | 211003 | Читаю 3, для підрахунку, яка це по рахунку цифра числа |
| 007 | WR R2 | 320002 | Записую в регістр |
| 008 | RD R1 | 310001 | Читаю номер числа |
| 009 | SUB #012 | 241012 | Віднімаю 12, для первірки |
| 010 | JZ 62 | 110062 | Якщо це вже 13 число – переходжу до обробки введеного масиву |
| 011 | IN 02 | 370002 | Якщо ні, заповнюю масив далі числами. Зчитую з клавіатури чи натиснута клавіша |
| 012 | JZ 8 | 110008 | Якщо ні, іду назад до зчитування чи натиснута клавіша |
| 013 | IN 00 | 370000 | Якщо натиснута, читаю код клавіші |
| 014 | WR R3 | 320003 | Записую в регістр код клавіші |
| 015 | OUT 10 | 380010 | Виводжу на екран |
| 016 | CALL 155 | 190155 | Очищаю клавіатуру |
| 017 | RD R2 | 310002 | Читаю яка це по рахунку цифра |
| 018 | SUB #003 | 241003 | Віднімаю 3 щоб перевірити, чи ця цифра сотні |
| 019 | JZ 26 | 110026 | Якщо так, іду до обробки цифри сотень |
| 020 | RD R2 | 310002 | Ящко ні, читаю розряд далі |
| 021 | SUB #002 | 241002 | Віднімаю 2 для перевірки |
| 022 | JZ 37 | 110037 | Якщо розряд десятки, іду обробляти десятки |
| 023 | RD R2 | 310002 | Якщо ж ні, іду читати значення розряду |
| 024 | SUB #001 | 241001 | Віднімаю 1 |
| 025 | JZ 48 | 110048 | Якщо розряд одиниці, іду обробляти одиниці |
| 026 | RD R0 | 310000 | Читаю початкову адресу масиву |
| 027 | ADD R1 | 330001 | Додаю до адреси порядок числа |
| 028 | WR R4 | 320004 | Записую в регістр на майбутнє |
| 029 | RD R3 | 310003 | Читаю код натиснутої клавіші |
| 030 | SUB #048 | 241048 | Відінмаю 48 щоб отримати цифру |
| 031 | MUL #100 | 251100 | Множу на 100 |
| 032 | WR @R4 | 324004 | Записую в адресу, що знаходиться в регістрі |
| 033 | RD R2 | 310002 | Читаю лічильник циклу |
| 034 | SUB #001 | 241001 | Віднімаю 1 щоб понизити розряд цифри з сотень до десятків |
| 035 | WR R2 | 320002 | Записую в регістр |
| 036 | JMP 8 | 100008 | Переходжу до очікування натиснення клавіші |
| 037 | RD R3 | 310003 | Обробка десятків. Читаю код натиснутої клавіші |
| 038 | SUB #048 | 241048 | Віднімаю 48 щоб отримати цифру |
| 039 | MUL #010 | 251010 | Множу на 10 |
| 040 | WR R5 | 320005 | Записую в регістр |
| 041 | RD @R4 | 314004 | Читаю число сотень із масиву для поточного числа |
| 042 | ADD R5 | 330005 | Додаю до нього число десятків |
| 043 | WR @R4 | 324004 | Записую назад в ОЗУ |
| 044 | RD R2 | 310002 | Читаю лічильник |
| 045 | SUB #001 | 241001 | Віднімаю 1 щоб понизити порядок |
| 046 | WR R2 | 320002 | Записую в регістр |
| 047 | JMP 8 | 100008 | Прееходжу до очікування натиснення на клавішу |
| 048 | RD R3 | 310003 | Обробка одиниць. Читаю код клавіші |
| 049 | SUB #048 | 241048 | Віднімаю 48 |
| 050 | WR R5 | 320005 | Записую в регістр |
| 051 | RD @R4 | 314004 | Читаю поточне число з числом сотень та десятків |
| 052 | ADD R5 | 330005 | Додаю розряд одиниць |
| 053 | WR @R4 | 324004 | Записую назад в ОЗУ |
| 054 | RD #003 | 211003 | Читаю 3, щоб скинути лічильник циклу для розрядів |
| 055 | WR R2 | 320002 | Записую в регістр |
| 056 | RD R1 | 310001 | Читаю число, яке це по рахунку число |
| 057 | ADD #001 | 231001 | Додаю 1 |
| 058 | WR R1 | 320001 | Записую в регістр |
| 059 | CALL 100 | 190100 | Викликаю таймер на 10 |
| 060 | CALL 148 | 190148 | Очищаю дисплей |
| 061 | JMP 8 | 100008 | Переходжу до очікування натиснення на клавішу |
| 062 | RD #000 | 211000 | Функція обробки масиву. Читаю число 0 щоб записати в лічильник для циклу |
| 063 | WR R2 | 320002 | Записую в регістр |
| 064 | RD #000 | 211000 | Читаю число 0 – кількість парних елементів |
| 065 | WR R9 | 320009 | Записую в регістр |
| 066 | RD R2 | 310002 | Читаю лічильник циклу |
| 067 | SUB #012 | 241012 | Віднімаю 12 щоб перевірити, чи оброблені вже всі числа в масиві |
| 068 | JZ 86 | 110086 | Якщо так, то переходжу до виводу результату |
| 069 | RD R0 | 310000 | Якщо ж ні, читаю початкову адресу масива |
| 070 | ADD R2 | 330002 | Додаю порядок поточного числа |
| 071 | WR R5 | 320005 | Записую в регістр |
| 072 | RD @R5 | 314005 | Читаю значення за адресою в регістрі |
| 073 | DIV #002 | 261002 | Ділю на 2 |
| 074 | WR R8 | 320008 | Записую в регістр |
| 075 | ADD R8 | 330008 | Додаю половину від числа |
| 076 | SUB @R5 | 344005 | Віднімаю від початкового числа для первірки, чи парне це число |
| 077 | JS 81 | 130081 | Якщо ні, іду до кінця циклу |
| 078 | RD R9 | 310009 | Якщо так, читаю кількість парних елементів |
| 079 | ADD #001 | 231001 | Додаю один |
| 080 | WR R9 | 320009 | Записую в регістр |
| 081 | RD R2 | 310002 | Кінець циклу. Читаю лічильник |
| 082 | ADD #001 | 231001 | Додаю 1 |
| 083 | WR R2 | 320002 | Записую в регістр |
| 084 | JMP 66 | 100066 | Іду до початку циклу до перевірки |
| 085 | CALL 81 | 190081 | Викликаємо кінець циклу |
| 086 | RD R9 | 310009 | Функція для виводу результату. Читаємо кількість парних елементів |
| 087 | DIV #100 | 261100 | Ділимо на 100 |
| 088 | MUL #100 | 251100 | Множимо на 100 |
| 089 | WR R0 | 320000 | Записуємо в регістр |
| 090 | RD R9 | 310009 | Читаємо кількість парних елементів |
| 091 | SUB R0 | 340000 | Віднімаємо число сотень |
| 092 | DIV #010 | 261010 | Ділимо на 10 |
| 093 | MUL #010 | 251010 | Множимо на 10 |
| 094 | WR R1 | 320001 | Записуємо в регістр |
| 095 | RD R9 | 310009 | Читаємо результат |
| 096 | SUB R1 | 340001 | Віднімаємо число сотень та десятків |
| 097 | SUB R0 | 340000 | Записуємо в регістр |
| 098 | WR R2 | 320002 | Переходжу до виводу результату на екран |
| 099 | JMP 120 | 100120 | функція для таймера на 10 од. зчитую число 10 |
| 100 | RD #010 | 211010 | вивантажую до таймера |
| 101 | OUT 21 | 380021 | читаю число 25101 для конфігурації таймера |
| 102 | RDI 025101 | 410000 025101 | виводжу в регістр параметрів таймера |
| 104 | OUT 22 | 380022 | підпрограмка для перевірки, чи закінчився час таймера. читаю значення з флагу про готовність у таймера |
| 105 | IN 20 | 370020 | якщо він 0, тобто час не закінчився, то переходжу до зчитування значення |
| 106 | JZ 105 | 110105 | якщо все ж час закінчився, читаю число 101 для скидання параметрів таймера, щоб він був готовий до роботи |
| 107 | RD #101 | 211101 | пересилаю ці параметри в таймер |
| 108 | OUT 20 | 380020 | кінець цієї підпрограмки |
| 109 | RET | 080000 | функція для таймера на 10 од. зчитую число 10 |
| 110 | RD #030 | 211030 | функція для таймера на 30 од. зчитую число 30 |
| 111 | OUT 21 | 380021 | вивантажую до таймера |
| 112 | RDI 025101 | 410000 025101 | читаю число 25101 для конфігурації таймера |
| 114 | OUT 22 | 380022 | виводжу в регістр параметрів таймера |
| 115 | IN 20 | 370020 | підпрограмка для перевірки, чи закінчився час таймера. читаю значення з флагу про готовність у таймера |
| 116 | JZ 105 | 110105 | якщо він 0, тобто час не закінчився, то переходжу до зчитування значення |
| 117 | RD #101 | 211101 | якщо все ж час закінчився, читаю число 101 для скидання параметрів таймера, щоб він був готовий до роботи |
| 118 | OUT 20 | 380020 | пересилаю ці параметри в таймер |
| 119 | RET | 080000 | кінець цієї підпрограмки |
| 120 | CALL 148 | 190148 | Функція виводу результату на екран. Очищаю дисплей |
| 121 | CALL 155 | 190155 | Очищаю буфер клавіатури |
| 122 | RD #082 | 211082 | Читаю символ |
| 123 | OUT 10 | 380010 | Виводжу на екран |
| 124 | RD #101 | 211101 | Читаю символ |
| 125 | OUT 10 | 380010 | Виводжу на екран |
| 126 | RD #115 | 211115 | Читаю символ |
| 127 | OUT 10 | 380010 | Виводжу на екран |
| 128 | RD #117 | 211117 | Читаю символ |
| 129 | OUT 10 | 380010 | Виводжу на екран |
| 130 | RD #108 | 211108 | Читаю символ |
| 131 | OUT 10 | 380010 | Виводжу на екран |
| 132 | RD #116 | 211116 | Читаю символ |
| 133 | OUT 10 | 380010 | Виводжу на екран |
| 134 | RD #058 | 211058 | Читаю символ |
| 135 | OUT 10 | 380010 | Виводжу на екран |
| 136 | RD #095 | 211095 | Читаю символ |
| 137 | OUT 10 | 380010 | Виводжу на екран |
| 138 | RD R0 | 310000 | Читаю число сотень |
| 139 | ADD #048 | 231048 | Додаю 48 |
| 140 | OUT 10 | 380010 | Виводжу на екран |
| 141 | RD R1 | 310001 | Читаю число десятків |
| 142 | ADD #048 | 231048 | Додаю 48 |
| 143 | OUT 10 | 380010 | Виводжу на екран |
| 144 | RD R2 | 310002 | Читаю число одиниць |
| 145 | ADD #048 | 231048 | Додаю 48 |
| 146 | OUT 10 | 380010 | Виводжу на екран |
| 147 | HLT | 090000 | Кінець програми |
| 148 | RD #101 | 211101 | функція очистки дисплея. читаю число 101 |
| 149 | OUT 11 | 380011 | відправляю в регістр 1 дисплея |
| 150 | RD #102 | 211102 | читаю число 102 |
| 151 | OUT 11 | 380011 | відправляю в регістр 1 дисплея |
| 152 | RD #011 | 211011 | читаю число 11 |
| 153 | OUT 11 | 380011 | відправляю в регістр 1 дисплея |
| 154 | RET | 080000 | кінець функції |
| 155 | RD #010 | 211010 | функція очистки клавіатури. читаю число 10 |
| 156 | OUT 01 | 380001 | відправляю в регістр параметрів клавіатури |
| 157 | RD #101 | 211101 | читаю число 101 |
| 158 | OUT 01 | 380001 | відправляю в регістр параметрів клавіатури |
| 159 | RD #103 | 211103 | читаю число 103 |
| 160 | OUT 01 | 380001 | відправляю в регістр параметрів клавіатури |
| 161 | RET | 080000 | кінець функції |

1. Надати у звіті разом із текстом програми скріншот роботи програми у моделі CPU.

