# Vd11:

## Bai11.c

|  |
| --- |
| #include<stdio.h>  int i,s=0,n;  main()  {    printf("Nhap n:");  scanf("%d",&n);  for(i=1;i<n;i++)  {  if(i%2!=0)  {  s=s+i\*i;  }  }  printf("Tong binh phuong cac so le be hon %d la: %d",n,s);  } |

## Bai11\_TU.c

|  |
| --- |
| #include<stdio.h>  void main()  {  int i,s=0,n;  printf("Nhap n:");  scanf("%d",&n);  for(i=1;i<n;i++)  {  if(i%2!=0)  {  s=s+i\*i;  }  }  printf("Tong binh phuong cac so le be hon %d la: %d",n,s);  } |

## Bai11\_8086.s

|  |
| --- |
| .file "b1.c"  .text  .comm i,4,4  .globl s  .bss  .align 4  .type s, @object  .size s, 4  s:  .zero 4  .comm n,4,4  .section .rodata  .LC0:  .string "Nhap n:"  .LC1:  .string "%d"  .align 8  .LC2:  .string "Tong binh phuong cac so le be hon %d la: %d"  .text  .globl main  .type main, @function  main:  .LFB0:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 6, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 6  leaq .LC0(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq n(%rip), %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl $1, i(%rip)  jmp .L2  .L4:  movl i(%rip), %eax  andl $1, %eax  testl %eax, %eax  je .L3  movl i(%rip), %edx  movl i(%rip), %eax  imull %eax, %edx  movl s(%rip), %eax  addl %edx, %eax  movl %eax, s(%rip)  .L3:  movl i(%rip), %eax  addl $1, %eax  movl %eax, i(%rip)  .L2:  movl i(%rip), %edx  movl n(%rip), %eax  cmpl %eax, %edx  jl .L4  movl s(%rip), %edx  movl n(%rip), %eax  movl %eax, %esi  leaq .LC2(%rip), %rdi  movl $0, %eax  call printf@PLT  movl $0, %eax  popq %rbp  .cfi\_def\_cfa 7, 8  ret  .cfi\_endproc  .LFE0:  .size main, .-main  .ident "GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",@progbits |

## Bai11\_8086\_TU.s

|  |
| --- |
| .file "b1.c"  .text  .comm i,4,4  .globl s  .bss  .align 4  .type s, @object  .size s, 4  s:  .zero 4  .comm n,4,4  .section .rodata  .LC0:  .string "Nhap n:"  .LC1:  .string "%d"  .align 8  .LC2:  .string "Tong binh phuong cac so le be hon %d la: %d"  .text  .globl main  .type main, @function  main:  .LFB0:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 1, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 1  leaq .LC0(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq n(%rip), %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl $1, i(%rip)  jmp .L2  .L4:  movl i(%rip), %eax  andl $1, %eax  testl %eax, %eax  je .L3  movl i(%rip), %edx  movl i(%rip), %eax  imull %eax, %edx  movl s(%rip), %eax  addl %edx, %eax  movl %eax, s(%rip)  .L3:  movl i(%rip), %eax  addl $1, %eax  movl %eax, i(%rip)  .L2:  movl i(%rip), %edx  movl n(%rip), %eax  cmpl %eax, %edx  jl .L4  movl s(%rip), %edx  movl n(%rip), %eax  movl %eax, %esi  leaq .LC2(%rip), %rdi  movl $0, %eax  call printf@PLT  movl $0, %eax  popq %rbp  .cfi\_def\_cfa 2, 8  ret  .cfi\_endproc  .LFE0:  .size main, .-main  .ident "GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",@progbits |

## Bai11\_ARM.s

|  |
| --- |
| .arch armv5t  .eabi\_attribute 20, 1  .eabi\_attribute 21, 1  .eabi\_attribute 23, 3  .eabi\_attribute 24, 1  .eabi\_attribute 25, 1  .eabi\_attribute 26, 2  .eabi\_attribute 30, 6  .eabi\_attribute 34, 0  .eabi\_attribute 18, 4  .file "b1.c"  .text  .comm i,4,4  .global s  .bss  .align 2  .type s, %object  .size s, 4  s:  .space 4  .comm n,4,4  .section .rodata  .align 2  .LC0:  .ascii "Nhap n:\000"  .align 2  .LC1:  .ascii "%d\000"  .align 2  .LC2:  .ascii "Tong binh phuong cac so le be hon %d la: %d\000"  .text  .align 2  .global main  .syntax unified  .arm  .fpu softvfp  .type main, %function  main:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 1, uses\_anonymous\_args = 0  push {fp, lr}  add fp, sp, #4  ldr r0, .L6  bl printf  ldr r1, .L6+4  ldr r0, .L6+8  bl \_\_isoc99\_scanf  ldr r3, .L6+12  mov r2, #1  str r2, [r3]  b .L2  .L4:  ldr r3, .L6+12  ldr r3, [r3]  and r3, r3, #1  cmp r3, #0  beq .L3  ldr r3, .L6+12  ldr r3, [r3]  ldr r2, .L6+12  ldr r1, [r2]  mul r2, r3, r1  ldr r3, .L6+16  ldr r3, [r3]  add r3, r2, r3  ldr r2, .L6+16  str r3, [r2]  .L3:  ldr r3, .L6+12  ldr r3, [r3]  add r3, r3, #1  ldr r2, .L6+12  str r3, [r2]  .L2:  ldr r3, .L6+12  ldr r2, [r3]  ldr r3, .L6+4  ldr r3, [r3]  cmp r2, r3  blt .L4  ldr r3, .L6+4  ldr r1, [r3]  ldr r3, .L6+16  ldr r3, [r3]  mov r2, r3  ldr r0, .L6+20  bl printf  mov r3, #0  mov r0, r3  pop {fp, pc}  .L7:  .align 2  .L6:  .word .LC0  .word n  .word .LC1  .word i  .word s  .word .LC2  .size main, .-main  .ident "GCC: (Ubuntu/Linaro 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",%progbits |

## Bai11\_ARM\_TU.s

|  |
| --- |
| .arch armv5t  .eabi\_attribute 20, 1  .eabi\_attribute 21, 1  .eabi\_attribute 23, 3  .eabi\_attribute 24, 1  .eabi\_attribute 25, 1  .eabi\_attribute 26, 2  .eabi\_attribute 30, 2  .eabi\_attribute 34, 0  .eabi\_attribute 18, 4  .file "b1.c"  .text  .section .rodata.str1.4,"aMS",%progbits,1  .align 2  .LC0:  .ascii "Nhap n:\000"  .align 2  .LC1:  .ascii "%d\000"  .align 2  .LC2:  .ascii "Tong binh phuong cac so le be hon %d la: %d\000"  .section .text.startup,"ax",%progbits  .align 2  .global main  .syntax unified  .arm  .fpu softvfp  .type main, %function  main:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 0, uses\_anonymous\_args = 0  push {r4, lr}  ldr r4, .L16  ldr r1, .L16+4  mov r0, #1  bl \_\_printf\_chk  mov r1, r4  ldr r0, .L16+8  bl \_\_isoc99\_scanf  mov r3, #1  ldr r2, [r4]  ldr lr, .L16+12  cmp r2, r3  ldrle ip, .L16+16  str r3, [lr]  ble .L3  mov r0, #0  ldr ip, .L16+16  ldr r1, [ip]  .L5:  tst r3, #1  mlane r1, r3, r3, r1  add r3, r3, #1  movne r0, #1  cmp r3, r2  bne .L5  cmp r0, #0  str r2, [lr]  strne r1, [ip]  .L3:  ldr r3, [ip]  ldr r1, .L16+20  mov r0, #1  bl \_\_printf\_chk  mov r0, #0  pop {r4, pc}  .L17:  .align 2  .L16:  .word n  .word .LC0  .word .LC1  .word i  .word .LANCHOR0  .word .LC2  .size main, .-main  .comm n,4,4  .global s  .comm i,4,4  .bss  .align 2  .set .LANCHOR0,. + 0  .type s, %object  .size s, 4  s:  .space 4  .ident "GCC: (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0"  .section .note.GNU-stack,"",%progbits |

# Vd12:

## Bai12.c

|  |
| --- |
| #include<stdio.h>  int a,b,c,max;  void nhap(){  printf("Nhap a=");  scanf("%d",&a);  printf("Nhap b=");  scanf("%d",&b);  printf("Nhap c=");  scanf("%d",&c);  }  void main()  {    nhap();  max=a;  if(b>max)  {  max=b;  }  if(c>max)  {  max=c;  }  printf("gia tri lon nhat la: %d",max);    } |

## Bai12\_TU.c

|  |
| --- |
| #include<stdio.h>  void main()  {  int a,b,c,max;  printf("Nhap a=");  scanf("%d",&a);  printf("Nhap b=");  scanf("%d",&b);  printf("Nhap c=");  scanf("%d",&c);  max=a;  if(b>max)  {  max=b;  }  if(c>max)  {  max=c;  }  printf("gia tri lon nhat la: %d",max);    } |

## Bai12\_8086.s

|  |
| --- |
| .file "b2.c"  .text  .comm a,4,4  .comm b,4,4  .comm c,4,4  .comm max,4,4  .section .rodata  .LC0:  .string "Nhap a="  .LC1:  .string "%d"  .LC2:  .string "Nhap b="  .LC3:  .string "Nhap c="  .text  .globl nhap  .type nhap, @function  nhap:  .LFB0:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 6, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 6  leaq .LC0(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq a(%rip), %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  leaq .LC2(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq b(%rip), %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  leaq .LC3(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq c(%rip), %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  nop  popq %rbp  .cfi\_def\_cfa 7, 8  ret  .cfi\_endproc  .LFE0:  .size nhap, .-nhap  .section .rodata  .LC4:  .string "gia tri lon nhat la: %d"  .text  .globl main  .type main, @function  main:  .LFB1:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 6, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 6  movl $0, %eax  call nhap  movl a(%rip), %eax  movl %eax, max(%rip)  movl b(%rip), %edx  movl max(%rip), %eax  cmpl %eax, %edx  jle .L3  movl b(%rip), %eax  movl %eax, max(%rip)  .L3:  movl c(%rip), %edx  movl max(%rip), %eax  cmpl %eax, %edx  jle .L4  movl c(%rip), %eax  movl %eax, max(%rip)  .L4:  movl max(%rip), %eax  movl %eax, %esi  leaq .LC4(%rip), %rdi  movl $0, %eax  call printf@PLT  nop  popq %rbp  .cfi\_def\_cfa 7, 8  ret  .cfi\_endproc  .LFE1:  .size main, .-main  .ident "GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",@progbits |

## Bai12\_8086\_TU.s

|  |
| --- |
| .file "b2.c"  .text  .comm a,4,4  .comm b,4,4  .comm c,4,4  .comm max,4,4  .section .rodata  .LC0:  .string "Nhap a="  .LC1:  .string "%d"  .LC2:  .string "Nhap b="  .LC3:  .string "Nhap c="  .text  .globl nhap  .type nhap, @function  nhap:  .LFB0:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 1, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 1  leaq .LC0(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq a(%rip), %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  leaq .LC2(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq b(%rip), %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  leaq .LC3(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq c(%rip), %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  nop  popq %rbp  .cfi\_def\_cfa 2, 8  ret  .cfi\_endproc  .LFE0:  .size nhap, .-nhap  .section .rodata  .LC4:  .string "gia tri lon nhat la: %d"  .text  .globl main  .type main, @function  main:  .LFB1:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 1, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 1  movl $0, %eax  call nhap  movl a(%rip), %eax  movl %eax, max(%rip)  movl b(%rip), %edx  movl max(%rip), %eax  cmpl %eax, %edx  jle .L3  movl b(%rip), %eax  movl %eax, max(%rip)  .L3:  movl c(%rip), %edx  movl max(%rip), %eax  cmpl %eax, %edx  jle .L4  movl c(%rip), %eax  movl %eax, max(%rip)  .L4:  movl max(%rip), %eax  movl %eax, %esi  leaq .LC4(%rip), %rdi  movl $0, %eax  call printf@PLT  nop  popq %rbp  .cfi\_def\_cfa 2, 8  ret  .cfi\_endproc  .LFE1:  .size main, .-main  .ident "GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",@progbits |

## Bai12\_ARM.s

|  |
| --- |
| .arch armv5t  .eabi\_attribute 20, 1  .eabi\_attribute 21, 1  .eabi\_attribute 23, 3  .eabi\_attribute 24, 1  .eabi\_attribute 25, 1  .eabi\_attribute 26, 2  .eabi\_attribute 30, 6  .eabi\_attribute 34, 0  .eabi\_attribute 18, 4  .file "b2.c"  .text  .comm a,4,4  .comm b,4,4  .comm c,4,4  .comm max,4,4  .section .rodata  .align 2  .LC0:  .ascii "Nhap a=\000"  .align 2  .LC1:  .ascii "%d\000"  .align 2  .LC2:  .ascii "Nhap b=\000"  .align 2  .LC3:  .ascii "Nhap c=\000"  .text  .align 2  .global nhap  .syntax unified  .arm  .fpu softvfp  .type nhap, %function  nhap:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 1, uses\_anonymous\_args = 0  push {fp, lr}  add fp, sp, #4  ldr r0, .L2  bl printf  ldr r1, .L2+4  ldr r0, .L2+8  bl \_\_isoc99\_scanf  ldr r0, .L2+12  bl printf  ldr r1, .L2+16  ldr r0, .L2+8  bl \_\_isoc99\_scanf  ldr r0, .L2+20  bl printf  ldr r1, .L2+24  ldr r0, .L2+8  bl \_\_isoc99\_scanf  nop  pop {fp, pc}  .L3:  .align 2  .L2:  .word .LC0  .word a  .word .LC1  .word .LC2  .word b  .word .LC3  .word c  .size nhap, .-nhap  .section .rodata  .align 2  .LC4:  .ascii "gia tri lon nhat la: %d\000"  .text  .align 2  .global main  .syntax unified  .arm  .fpu softvfp  .type main, %function  main:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 1, uses\_anonymous\_args = 0  push {fp, lr}  add fp, sp, #4  bl nhap  ldr r3, .L7  ldr r3, [r3]  ldr r2, .L7+4  str r3, [r2]  ldr r3, .L7+8  ldr r2, [r3]  ldr r3, .L7+4  ldr r3, [r3]  cmp r2, r3  ble .L5  ldr r3, .L7+8  ldr r3, [r3]  ldr r2, .L7+4  str r3, [r2]  .L5:  ldr r3, .L7+12  ldr r2, [r3]  ldr r3, .L7+4  ldr r3, [r3]  cmp r2, r3  ble .L6  ldr r3, .L7+12  ldr r3, [r3]  ldr r2, .L7+4  str r3, [r2]  .L6:  ldr r3, .L7+4  ldr r3, [r3]  mov r1, r3  ldr r0, .L7+16  bl printf  nop  pop {fp, pc}  .L8:  .align 2  .L7:  .word a  .word max  .word b  .word c  .word .LC4  .size main, .-main  .ident "GCC: (Ubuntu/Linaro 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",%progbits |

## Bai12\_ARM\_TU.s

|  |
| --- |
| .arch armv5t  .eabi\_attribute 20, 1  .eabi\_attribute 21, 1  .eabi\_attribute 23, 3  .eabi\_attribute 24, 1  .eabi\_attribute 25, 1  .eabi\_attribute 26, 2  .eabi\_attribute 30, 2  .eabi\_attribute 34, 0  .eabi\_attribute 18, 4  .file "b2.c"  .text  .section .rodata.str1.4,"aMS",%progbits,1  .align 2  .LC0:  .ascii "Nhap a=\000"  .align 2  .LC1:  .ascii "%d\000"  .align 2  .LC2:  .ascii "Nhap b=\000"  .align 2  .LC3:  .ascii "Nhap c=\000"  .text  .align 2  .global nhap  .syntax unified  .arm  .fpu softvfp  .type nhap, %function  nhap:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 0, uses\_anonymous\_args = 0  push {r4, lr}  ldr r4, .L4  ldr r1, .L4+4  mov r0, #1  bl \_\_printf\_chk  mov r0, r4  ldr r1, .L4+8  bl \_\_isoc99\_scanf  ldr r1, .L4+12  mov r0, #1  bl \_\_printf\_chk  mov r0, r4  ldr r1, .L4+16  bl \_\_isoc99\_scanf  ldr r1, .L4+20  mov r0, #1  bl \_\_printf\_chk  mov r0, r4  pop {r4, lr}  ldr r1, .L4+24  b \_\_isoc99\_scanf  .L5:  .align 2  .L4:  .word .LC1  .word .LC0  .word a  .word .LC2  .word b  .word .LC3  .word c  .size nhap, .-nhap  .section .rodata.str1.4  .align 2  .LC4:  .ascii "gia tri lon nhat la: %d\000"  .section .text.startup,"ax",%progbits  .align 2  .global main  .syntax unified  .arm  .fpu softvfp  .type main, %function  main:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 0, uses\_anonymous\_args = 0  push {r4, lr}  bl nhap  pop {r4, lr}  ldr r3, .L10  ldr r2, .L10+4  ldr r3, [r3]  ldr r2, [r2]  ldr r1, .L10+8  cmp r2, r3  movge r3, r2  ldr r2, [r1]  ldr ip, .L10+12  cmp r2, r3  movlt r2, r3  mov r0, #1  ldr r1, .L10+16  str r2, [ip]  b \_\_printf\_chk  .L11:  .align 2  .L10:  .word a  .word b  .word c  .word max  .word .LC4  .size main, .-main  .comm max,4,4  .comm c,4,4  .comm b,4,4  .comm a,4,4  .ident "GCC: (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0"  .section .note.GNU-stack,"",%progbits |

# Vd13:

## Bai13.c

|  |
| --- |
| #include<stdio.h>  int x;  main()  {    printf("Nhap x:");  scanf("%d",&x);  if(x%2==0)  {  printf("%d la so chan",x);  }  else  {  printf("%d la so le",x);  }  } |

## Bai13\_TU.c

|  |
| --- |
| #include<stdio.h>  void main()  {  int x;  printf("Nhap x:");  scanf("%d",&x);  if(x%2==0)  {  printf("%d la so chan",x);  }  else  {  printf("%d la so le",x);  }  } |

## Bai13\_8086.s

|  |
| --- |
| .file "b3.c"  .text  .comm x,4,4  .section .rodata  .LC0:  .string "Nhap x:"  .LC1:  .string "%d"  .LC2:  .string "%d la so chan"  .LC3:  .string "%d la so le"  .text  .globl main  .type main, @function  main:  .LFB0:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 6, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 6  leaq .LC0(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq x(%rip), %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl x(%rip), %eax  andl $1, %eax  testl %eax, %eax  jne .L2  movl x(%rip), %eax  movl %eax, %esi  leaq .LC2(%rip), %rdi  movl $0, %eax  call printf@PLT  jmp .L3  .L2:  movl x(%rip), %eax  movl %eax, %esi  leaq .LC3(%rip), %rdi  movl $0, %eax  call printf@PLT  .L3:  movl $0, %eax  popq %rbp  .cfi\_def\_cfa 7, 8  ret  .cfi\_endproc  .LFE0:  .size main, .-main  .ident "GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",@progbits |

## Bai13\_8086\_TU.s

|  |
| --- |
| .file "b3.c"  .text  .comm x,4,4  .section .rodata  .LC0:  .string "Nhap x:"  .LC1:  .string "%d"  .LC2:  .string "%d la so chan"  .LC3:  .string "%d la so le"  .text  .globl main  .type main, @function  main:  .LFB0:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 1, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 1  leaq .LC0(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq x(%rip), %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl x(%rip), %eax  andl $1, %eax  testl %eax, %eax  jne .L2  movl x(%rip), %eax  movl %eax, %esi  leaq .LC2(%rip), %rdi  movl $0, %eax  call printf@PLT  jmp .L3  .L2:  movl x(%rip), %eax  movl %eax, %esi  leaq .LC3(%rip), %rdi  movl $0, %eax  call printf@PLT  .L3:  movl $0, %eax  popq %rbp  .cfi\_def\_cfa 2, 8  ret  .cfi\_endproc  .LFE0:  .size main, .-main  .ident "GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",@progbits |

## Bai13\_ARM.s

|  |
| --- |
| .arch armv5t  .eabi\_attribute 20, 1  .eabi\_attribute 21, 1  .eabi\_attribute 23, 3  .eabi\_attribute 24, 1  .eabi\_attribute 25, 1  .eabi\_attribute 26, 2  .eabi\_attribute 30, 6  .eabi\_attribute 34, 0  .eabi\_attribute 18, 4  .file "b3.c"  .text  .comm x,4,4  .section .rodata  .align 2  .LC0:  .ascii "Nhap x:\000"  .align 2  .LC1:  .ascii "%d\000"  .align 2  .LC2:  .ascii "%d la so chan\000"  .align 2  .LC3:  .ascii "%d la so le\000"  .text  .align 2  .global main  .syntax unified  .arm  .fpu softvfp  .type main, %function  main:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 1, uses\_anonymous\_args = 0  push {fp, lr}  add fp, sp, #4  ldr r0, .L5  bl printf  ldr r1, .L5+4  ldr r0, .L5+8  bl \_\_isoc99\_scanf  ldr r3, .L5+4  ldr r3, [r3]  and r3, r3, #1  cmp r3, #0  bne .L2  ldr r3, .L5+4  ldr r3, [r3]  mov r1, r3  ldr r0, .L5+12  bl printf  b .L3  .L2:  ldr r3, .L5+4  ldr r3, [r3]  mov r1, r3  ldr r0, .L5+16  bl printf  .L3:  mov r3, #0  mov r0, r3  pop {fp, pc}  .L6:  .align 2  .L5:  .word .LC0  .word x  .word .LC1  .word .LC2  .word .LC3  .size main, .-main  .ident "GCC: (Ubuntu/Linaro 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",%progbits |

## Bai13\_ARM\_TU.s

|  |
| --- |
| .arch armv5t  .eabi\_attribute 20, 1  .eabi\_attribute 21, 1  .eabi\_attribute 23, 3  .eabi\_attribute 24, 1  .eabi\_attribute 25, 1  .eabi\_attribute 26, 2  .eabi\_attribute 30, 2  .eabi\_attribute 34, 0  .eabi\_attribute 18, 4  .file "b3.c"  .text  .section .rodata.str1.4,"aMS",%progbits,1  .align 2  .LC0:  .ascii "Nhap x:\000"  .align 2  .LC1:  .ascii "%d\000"  .align 2  .LC2:  .ascii "%d la so chan\000"  .align 2  .LC3:  .ascii "%d la so le\000"  .section .text.startup,"ax",%progbits  .align 2  .global main  .syntax unified  .arm  .fpu softvfp  .type main, %function  main:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 0, uses\_anonymous\_args = 0  push {r4, lr}  ldr r4, .L6  ldr r1, .L6+4  mov r0, #1  bl \_\_printf\_chk  mov r1, r4  ldr r0, .L6+8  bl \_\_isoc99\_scanf  ldr r2, [r4]  mov r0, #1  tst r2, #1  ldreq r1, .L6+12  ldrne r1, .L6+16  bl \_\_printf\_chk  mov r0, #0  pop {r4, pc}  .L7:  .align 2  .L6:  .word x  .word .LC0  .word .LC1  .word .LC2  .word .LC3  .size main, .-main  .comm x,4,4  .ident "GCC: (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0"  .section .note.GNU-stack,"",%progbits |

# Vd14:

## Bai14.c

|  |
| --- |
| #include<stdio.h>  int x,y,a,b;  main()  {    do  {  printf("Nhap a,b = ");  scanf("%d%d",&a,&b);  }  while(a<=0 || b<=0);  x=a;  y=b;  while(a!=b)  {  if(a>b)  {  a-=b;  }  else  {  b-=a;  }  }  printf("Uoc chung lon nhat la %d",a);  printf("\nBoi chung nho nhat la %d",(x\*y)/a);    } |

## Bai14\_TU.c

|  |
| --- |
| #include<stdio.h>  void main()  {  int x,y,a,b;  do  {  printf("Nhap a,b = ");  scanf("%d%d",&a,&b);  }  while(a<=0 || b<=0);  x=a;  y=b;  while(a!=b)  {  if(a>b)  {  a-=b;  }  else  {  b-=a;  }  }  int bc = (x\*y)/a;  printf("Uoc chung lon nhat la %d",a);  printf("\nBoi chung nho nhat la %d",bc);    } |

## Bai14\_8086.s

|  |
| --- |
| .file "b4.c"  .text  .comm x,4,4  .comm y,4,4  .comm a,4,4  .comm b,4,4  .section .rodata  .LC0:  .string "Nhap a,b = "  .LC1:  .string "%d%d"  .LC2:  .string "Uoc chung lon nhat la %d"  .LC3:  .string "\nBoi chung nho nhat la %d"  .text  .globl main  .type main, @function  main:  .LFB0:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 6, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 6  .L2:  leaq .LC0(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq b(%rip), %rdx  leaq a(%rip), %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl a(%rip), %eax  testl %eax, %eax  jle .L2  movl b(%rip), %eax  testl %eax, %eax  jle .L2  movl a(%rip), %eax  movl %eax, x(%rip)  movl b(%rip), %eax  movl %eax, y(%rip)  jmp .L3  .L5:  movl a(%rip), %edx  movl b(%rip), %eax  cmpl %eax, %edx  jle .L4  movl a(%rip), %edx  movl b(%rip), %eax  subl %eax, %edx  movl %edx, %eax  movl %eax, a(%rip)  jmp .L3  .L4:  movl b(%rip), %edx  movl a(%rip), %eax  subl %eax, %edx  movl %edx, %eax  movl %eax, b(%rip)  .L3:  movl a(%rip), %edx  movl b(%rip), %eax  cmpl %eax, %edx  jne .L5  movl a(%rip), %eax  movl %eax, %esi  leaq .LC2(%rip), %rdi  movl $0, %eax  call printf@PLT  movl x(%rip), %edx  movl y(%rip), %eax  imull %edx, %eax  movl a(%rip), %ecx  cltd  idivl %ecx  movl %eax, %esi  leaq .LC3(%rip), %rdi  movl $0, %eax  call printf@PLT  movl $0, %eax  popq %rbp  .cfi\_def\_cfa 7, 8  ret  .cfi\_endproc  .LFE0:  .size main, .-main  .ident "GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",@progbits |

## Bai14\_8086\_TU.s

|  |
| --- |
| .file "b4.c"  .text  .comm x,4,4  .comm y,4,4  .comm a,4,4  .comm b,4,4  .section .rodata  .LC0:  .string "Nhap a,b = "  .LC1:  .string "%d%d"  .LC2:  .string "Uoc chung lon nhat la %d"  .LC3:  .string "\nBoi chung nho nhat la %d"  .text  .globl main  .type main, @function  main:  .LFB0:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 1, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 1  .L2:  leaq .LC0(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq b(%rip), %rdx  leaq a(%rip), %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl a(%rip), %eax  testl %eax, %eax  jle .L2  movl b(%rip), %eax  testl %eax, %eax  jle .L2  movl a(%rip), %eax  movl %eax, x(%rip)  movl b(%rip), %eax  movl %eax, y(%rip)  jmp .L3  .L5:  movl a(%rip), %edx  movl b(%rip), %eax  cmpl %eax, %edx  jle .L4  movl a(%rip), %edx  movl b(%rip), %eax  subl %eax, %edx  movl %edx, %eax  movl %eax, a(%rip)  jmp .L3  .L4:  movl b(%rip), %edx  movl a(%rip), %eax  subl %eax, %edx  movl %edx, %eax  movl %eax, b(%rip)  .L3:  movl a(%rip), %edx  movl b(%rip), %eax  cmpl %eax, %edx  jne .L5  movl a(%rip), %eax  movl %eax, %esi  leaq .LC2(%rip), %rdi  movl $0, %eax  call printf@PLT  movl x(%rip), %edx  movl y(%rip), %eax  imull %edx, %eax  movl a(%rip), %ecx  cltd  idivl %ecx  movl %eax, %esi  leaq .LC3(%rip), %rdi  movl $0, %eax  call printf@PLT  movl $0, %eax  popq %rbp  .cfi\_def\_cfa 2, 8  ret  .cfi\_endproc  .LFE0:  .size main, .-main  .ident "GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",@progbits |

## Bai14\_ARM.s

|  |
| --- |
| .arch armv5t  .eabi\_attribute 20, 1  .eabi\_attribute 21, 1  .eabi\_attribute 23, 3  .eabi\_attribute 24, 1  .eabi\_attribute 25, 1  .eabi\_attribute 26, 2  .eabi\_attribute 30, 6  .eabi\_attribute 34, 0  .eabi\_attribute 18, 4  .file "b4.c"  .text  .comm x,4,4  .comm y,4,4  .comm a,4,4  .comm b,4,4  .section .rodata  .align 2  .LC0:  .ascii "Nhap a,b = \000"  .align 2  .LC1:  .ascii "%d%d\000"  .align 2  .LC2:  .ascii "Uoc chung lon nhat la %d\000"  .global \_\_aeabi\_idiv  .align 2  .LC3:  .ascii "\012Boi chung nho nhat la %d\000"  .text  .align 2  .global main  .syntax unified  .arm  .fpu softvfp  .type main, %function  main:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 1, uses\_anonymous\_args = 0  push {fp, lr}  add fp, sp, #4  .L2:  ldr r0, .L7  bl printf  ldr r2, .L7+4  ldr r1, .L7+8  ldr r0, .L7+12  bl \_\_isoc99\_scanf  ldr r3, .L7+8  ldr r3, [r3]  cmp r3, #0  ble .L2  ldr r3, .L7+4  ldr r3, [r3]  cmp r3, #0  ble .L2  ldr r3, .L7+8  ldr r3, [r3]  ldr r2, .L7+16  str r3, [r2]  ldr r3, .L7+4  ldr r3, [r3]  ldr r2, .L7+20  str r3, [r2]  b .L3  .L5:  ldr r3, .L7+8  ldr r2, [r3]  ldr r3, .L7+4  ldr r3, [r3]  cmp r2, r3  ble .L4  ldr r3, .L7+8  ldr r2, [r3]  ldr r3, .L7+4  ldr r3, [r3]  sub r3, r2, r3  ldr r2, .L7+8  str r3, [r2]  b .L3  .L4:  ldr r3, .L7+4  ldr r2, [r3]  ldr r3, .L7+8  ldr r3, [r3]  sub r3, r2, r3  ldr r2, .L7+4  str r3, [r2]  .L3:  ldr r3, .L7+8  ldr r2, [r3]  ldr r3, .L7+4  ldr r3, [r3]  cmp r2, r3  bne .L5  ldr r3, .L7+8  ldr r3, [r3]  mov r1, r3  ldr r0, .L7+24  bl printf  ldr r3, .L7+16  ldr r3, [r3]  ldr r2, .L7+20  ldr r1, [r2]  mul r2, r3, r1  ldr r3, .L7+8  ldr r3, [r3]  mov r1, r3  mov r0, r2  bl \_\_aeabi\_idiv  mov r3, r0  mov r1, r3  ldr r0, .L7+28  bl printf  mov r3, #0  mov r0, r3  pop {fp, pc}  .L8:  .align 2  .L7:  .word .LC0  .word b  .word a  .word .LC1  .word x  .word y  .word .LC2  .word .LC3  .size main, .-main  .ident "GCC: (Ubuntu/Linaro 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",%progbits |

## Bai14\_ARM\_TU.s

|  |
| --- |
| .arch armv5t  .eabi\_attribute 20, 1  .eabi\_attribute 21, 1  .eabi\_attribute 23, 3  .eabi\_attribute 24, 1  .eabi\_attribute 25, 1  .eabi\_attribute 26, 2  .eabi\_attribute 30, 2  .eabi\_attribute 34, 0  .eabi\_attribute 18, 4  .file "b4.c"  .text  .section .rodata.str1.4,"aMS",%progbits,1  .align 2  .LC0:  .ascii "Nhap a,b = \000"  .align 2  .LC1:  .ascii "%d%d\000"  .align 2  .LC2:  .ascii "Uoc chung lon nhat la %d\000"  .global \_\_aeabi\_idiv  .align 2  .LC3:  .ascii "\012Boi chung nho nhat la %d\000"  .section .text.startup,"ax",%progbits  .align 2  .global main  .syntax unified  .arm  .fpu softvfp  .type main, %function  main:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 0, uses\_anonymous\_args = 0  push {r4, r5, r6, r7, r8, r9, r10, lr}  ldr r9, .L20  ldr r4, .L20+4  mov r8, r9  mov r7, r4  ldr r6, .L20+8  ldr r5, .L20+12  .L15:  mov r1, r6  mov r0, #1  bl \_\_printf\_chk  mov r2, r8  mov r1, r7  mov r0, r5  bl \_\_isoc99\_scanf  ldr r3, [r4]  cmp r3, #0  ble .L15  ldr r2, [r9]  cmp r2, #0  ble .L15  ldr r6, .L20+16  ldr r5, .L20+20  cmp r3, r2  str r3, [r6]  str r2, [r5]  beq .L3  mov r1, #0  mov r0, r1  .L6:  cmp r2, r3  sublt r3, r3, r2  subge r2, r2, r3  movlt r0, #1  movge r1, #1  cmp r3, r2  bne .L6  cmp r0, #0  strne r2, [r4]  cmp r1, #0  strne r2, [r9]  .L3:  ldr r1, .L20+24  mov r0, #1  bl \_\_printf\_chk  ldr r3, [r6]  ldr r0, [r5]  ldr r1, [r4]  mul r0, r3, r0  bl \_\_aeabi\_idiv  ldr r1, .L20+28  mov r2, r0  mov r0, #1  bl \_\_printf\_chk  mov r0, #0  pop {r4, r5, r6, r7, r8, r9, r10, pc}  .L21:  .align 2  .L20:  .word b  .word a  .word .LC0  .word .LC1  .word x  .word y  .word .LC2  .word .LC3  .size main, .-main  .comm b,4,4  .comm a,4,4  .comm y,4,4  .comm x,4,4  .ident "GCC: (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0"  .section .note.GNU-stack,"",%progbits |

# Vd15:

## Bai15.c

|  |
| --- |
| #include<stdio.h>  #include<math.h>  int x;  int s=0,i;  void nhap(){  printf("Nhap mot so nguyen duong \n");  scanf("%d", &x);  }  main()  {  nhap();  for(i=1;i<x;i++)  {  if(x%i== 0)  {  s=s+i;  }  }  if(s==x)  {  printf("%d la so hoan hao",x);  }  else  {  printf("%d khong phai la so hoan hao",x);  }    } |

## Bai15\_TU.c

|  |
| --- |
| #include<stdio.h>  #include<math.h>  void main()  {  int x;  int s=0,i;  printf("Nhap mot so nguyen duong \n");  scanf("%d", &x);  for(i=1;i<x;i++)  {  if(x%i== 0)  {  s=s+i;  }  }  if(s==x)  {  printf("%d la so hoan hao",x);  }  else  {  printf("%d khong phai la so hoan hao",x);  }    } |

## Bai15\_8086.s

|  |
| --- |
| .file "b5.c"  .text  .comm x,4,4  .globl s  .bss  .align 4  .type s, @object  .size s, 4  s:  .zero 4  .comm i,4,4  .section .rodata  .LC0:  .string "Nhap mot so nguyen duong "  .LC1:  .string "%d"  .text  .globl nhap  .type nhap, @function  nhap:  .LFB0:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 6, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 6  leaq .LC0(%rip), %rdi  call puts@PLT  leaq x(%rip), %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  nop  popq %rbp  .cfi\_def\_cfa 7, 8  ret  .cfi\_endproc  .LFE0:  .size nhap, .-nhap  .section .rodata  .LC2:  .string "%d la so hoan hao"  .LC3:  .string "%d khong phai la so hoan hao"  .text  .globl main  .type main, @function  main:  .LFB1:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 6, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 6  movl $0, %eax  call nhap  movl $1, i(%rip)  jmp .L3  .L5:  movl x(%rip), %eax  movl i(%rip), %ecx  cltd  idivl %ecx  movl %edx, %eax  testl %eax, %eax  jne .L4  movl s(%rip), %edx  movl i(%rip), %eax  addl %edx, %eax  movl %eax, s(%rip)  .L4:  movl i(%rip), %eax  addl $1, %eax  movl %eax, i(%rip)  .L3:  movl i(%rip), %edx  movl x(%rip), %eax  cmpl %eax, %edx  jl .L5  movl s(%rip), %edx  movl x(%rip), %eax  cmpl %eax, %edx  jne .L6  movl x(%rip), %eax  movl %eax, %esi  leaq .LC2(%rip), %rdi  movl $0, %eax  call printf@PLT  jmp .L7  .L6:  movl x(%rip), %eax  movl %eax, %esi  leaq .LC3(%rip), %rdi  movl $0, %eax  call printf@PLT  .L7:  movl $0, %eax  popq %rbp  .cfi\_def\_cfa 7, 8  ret  .cfi\_endproc  .LFE1:  .size main, .-main  .ident "GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",@progbits |

## Bai15\_8086\_TU.s

|  |
| --- |
| .file "b5.c"  .text  .comm x,4,4  .globl s  .bss  .align 4  .type s, @object  .size s, 4  s:  .zero 4  .comm i,4,4  .section .rodata  .LC0:  .string "Nhap mot so nguyen duong "  .LC1:  .string "%d"  .text  .globl nhap  .type nhap, @function  nhap:  .LFB0:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 1, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 1  leaq .LC0(%rip), %rdi  call puts@PLT  leaq x(%rip), %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  nop  popq %rbp  .cfi\_def\_cfa 2, 8  ret  .cfi\_endproc  .LFE0:  .size nhap, .-nhap  .section .rodata  .LC2:  .string "%d la so hoan hao"  .LC3:  .string "%d khong phai la so hoan hao"  .text  .globl main  .type main, @function  main:  .LFB1:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 1, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 1  movl $0, %eax  call nhap  movl $1, i(%rip)  jmp .L3  .L5:  movl x(%rip), %eax  movl i(%rip), %ecx  cltd  idivl %ecx  movl %edx, %eax  testl %eax, %eax  jne .L4  movl s(%rip), %edx  movl i(%rip), %eax  addl %edx, %eax  movl %eax, s(%rip)  .L4:  movl i(%rip), %eax  addl $1, %eax  movl %eax, i(%rip)  .L3:  movl i(%rip), %edx  movl x(%rip), %eax  cmpl %eax, %edx  jl .L5  movl s(%rip), %edx  movl x(%rip), %eax  cmpl %eax, %edx  jne .L6  movl x(%rip), %eax  movl %eax, %esi  leaq .LC2(%rip), %rdi  movl $0, %eax  call printf@PLT  jmp .L7  .L6:  movl x(%rip), %eax  movl %eax, %esi  leaq .LC3(%rip), %rdi  movl $0, %eax  call printf@PLT  .L7:  movl $0, %eax  popq %rbp  .cfi\_def\_cfa 2, 8  ret  .cfi\_endproc  .LFE1:  .size main, .-main  .ident "GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",@progbits |

## Bai15\_ARM.s

|  |
| --- |
| .arch armv5t  .eabi\_attribute 20, 1  .eabi\_attribute 21, 1  .eabi\_attribute 23, 3  .eabi\_attribute 24, 1  .eabi\_attribute 25, 1  .eabi\_attribute 26, 2  .eabi\_attribute 30, 6  .eabi\_attribute 34, 0  .eabi\_attribute 18, 4  .file "b5.c"  .text  .comm x,4,4  .global s  .bss  .align 2  .type s, %object  .size s, 4  s:  .space 4  .comm i,4,4  .section .rodata  .align 2  .LC0:  .ascii "Nhap mot so nguyen duong \000"  .align 2  .LC1:  .ascii "%d\000"  .text  .align 2  .global nhap  .syntax unified  .arm  .fpu softvfp  .type nhap, %function  nhap:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 1, uses\_anonymous\_args = 0  push {fp, lr}  add fp, sp, #4  ldr r0, .L2  bl puts  ldr r1, .L2+4  ldr r0, .L2+8  bl \_\_isoc99\_scanf  nop  pop {fp, pc}  .L3:  .align 2  .L2:  .word .LC0  .word x  .word .LC1  .size nhap, .-nhap  .global \_\_aeabi\_idivmod  .section .rodata  .align 2  .LC2:  .ascii "%d la so hoan hao\000"  .align 2  .LC3:  .ascii "%d khong phai la so hoan hao\000"  .text  .align 2  .global main  .syntax unified  .arm  .fpu softvfp  .type main, %function  main:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 1, uses\_anonymous\_args = 0  push {fp, lr}  add fp, sp, #4  bl nhap  ldr r3, .L11  mov r2, #1  str r2, [r3]  b .L5  .L7:  ldr r3, .L11+4  ldr r2, [r3]  ldr r3, .L11  ldr r3, [r3]  mov r1, r3  mov r0, r2  bl \_\_aeabi\_idivmod  mov r3, r1  cmp r3, #0  bne .L6  ldr r3, .L11+8  ldr r2, [r3]  ldr r3, .L11  ldr r3, [r3]  add r3, r2, r3  ldr r2, .L11+8  str r3, [r2]  .L6:  ldr r3, .L11  ldr r3, [r3]  add r3, r3, #1  ldr r2, .L11  str r3, [r2]  .L5:  ldr r3, .L11  ldr r2, [r3]  ldr r3, .L11+4  ldr r3, [r3]  cmp r2, r3  blt .L7  ldr r3, .L11+8  ldr r2, [r3]  ldr r3, .L11+4  ldr r3, [r3]  cmp r2, r3  bne .L8  ldr r3, .L11+4  ldr r3, [r3]  mov r1, r3  ldr r0, .L11+12  bl printf  b .L9  .L8:  ldr r3, .L11+4  ldr r3, [r3]  mov r1, r3  ldr r0, .L11+16  bl printf  .L9:  mov r3, #0  mov r0, r3  pop {fp, pc}  .L12:  .align 2  .L11:  .word i  .word x  .word s  .word .LC2  .word .LC3  .size main, .-main  .ident "GCC: (Ubuntu/Linaro 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",%progbits |

## Bai15\_ARM\_TU.s

|  |
| --- |
| .arch armv5t  .eabi\_attribute 20, 1  .eabi\_attribute 21, 1  .eabi\_attribute 23, 3  .eabi\_attribute 24, 1  .eabi\_attribute 25, 1  .eabi\_attribute 26, 2  .eabi\_attribute 30, 2  .eabi\_attribute 34, 0  .eabi\_attribute 18, 4  .file "b5.c"  .text  .section .rodata.str1.4,"aMS",%progbits,1  .align 2  .LC0:  .ascii "Nhap mot so nguyen duong \000"  .align 2  .LC1:  .ascii "%d\000"  .text  .align 2  .global nhap  .syntax unified  .arm  .fpu softvfp  .type nhap, %function  nhap:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 0, uses\_anonymous\_args = 0  push {r4, lr}  ldr r0, .L4  bl puts  pop {r4, lr}  ldr r1, .L4+4  ldr r0, .L4+8  b \_\_isoc99\_scanf  .L5:  .align 2  .L4:  .word .LC0  .word x  .word .LC1  .size nhap, .-nhap  .global \_\_aeabi\_idivmod  .section .rodata.str1.4  .align 2  .LC2:  .ascii "%d la so hoan hao\000"  .align 2  .LC3:  .ascii "%d khong phai la so hoan hao\000"  .section .text.startup,"ax",%progbits  .align 2  .global main  .syntax unified  .arm  .fpu softvfp  .type main, %function  main:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 0, uses\_anonymous\_args = 0  push {r4, r5, r6, r7, r8, r9, r10, lr}  mov r4, #1  bl nhap  ldr r3, .L19  ldr r9, .L19+4  ldr r5, [r3]  str r4, [r9]  cmp r5, r4  ldrle r8, .L19+8  ble .L8  mov r7, #0  ldr r8, .L19+8  ldr r6, [r8]  .L10:  mov r1, r4  mov r0, r5  bl \_\_aeabi\_idivmod  cmp r1, #0  addeq r6, r6, r4  add r4, r4, #1  moveq r7, #1  cmp r5, r4  bne .L10  cmp r7, #0  str r5, [r9]  strne r6, [r8]  .L8:  ldr r3, [r8]  mov r2, r5  cmp r3, r5  ldreq r1, .L19+12  ldrne r1, .L19+16  mov r0, #1  bl \_\_printf\_chk  mov r0, #0  pop {r4, r5, r6, r7, r8, r9, r10, pc}  .L20:  .align 2  .L19:  .word x  .word i  .word .LANCHOR0  .word .LC2  .word .LC3  .size main, .-main  .comm i,4,4  .global s  .comm x,4,4  .bss  .align 2  .set .LANCHOR0,. + 0  .type s, %object  .size s, 4  s:  .space 4  .ident "GCC: (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0"  .section .note.GNU-stack,"",%progbits |

# Vd16:

## Bai16.c

|  |
| --- |
| #include<stdio.h>  void NhapMang(int a[],int n)  {  int i;  for(i=0;i<n;i++)  {  printf("Nhap a[%d]: ",i);  scanf("%d",&a[i]);  }  }  void In(int a[],int n)  {  int i;  int s=0;  printf("Tong cac so le trong mang la: \n");  for(i=0;i<n;i++)  {  s+=a[i];  }  printf("%d",s);  }  int main()  {  int n;  int a[n];  printf("Nhap n: ");  scanf("%d",&n);  NhapMang(a, n);  In(a, n);  } |

## Bai16\_TU.c

|  |
| --- |
| #include<stdio.h>  void main()  {  int a[50];  int i,n,s=0;  printf("Nhap so phan tu mang: ");  scanf("%d",&n);  for(i=0;i<n;i++)  {  scanf("%d",&a[i]);  }    for(i=0;i<n;i++)  {  s +=a[i];  }  printf("\nTong Mang vua nhap la : %d",s);    } int n=5;    int dem = 0;  int i = 2;  while (dem < n) {  if (isPrimeNumber(i)) {  printf("%d ", i);  dem++;  }  i++;  }  } |

## Bai16\_8086.s

|  |
| --- |
| .file "b6.c"  .text  .section .rodata  .LC0:  .string "Nhap a[%d]: "  .LC1:  .string "%d"  .text  .globl NhapMang  .type NhapMang, @function  NhapMang:  .LFB0:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 6, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 6  subq $32, %rsp  movq %rdi, -24(%rbp)  movl %esi, -28(%rbp)  movl $0, -4(%rbp)  jmp .L2  .L3:  movl -4(%rbp), %eax  movl %eax, %esi  leaq .LC0(%rip), %rdi  movl $0, %eax  call printf@PLT  movl -4(%rbp), %eax  cltq  leaq 0(,%rax,4), %rdx  movq -24(%rbp), %rax  addq %rdx, %rax  movq %rax, %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  addl $1, -4(%rbp)  .L2:  movl -4(%rbp), %eax  cmpl -28(%rbp), %eax  jl .L3  nop  leave  .cfi\_def\_cfa 7, 8  ret  .cfi\_endproc  .LFE0:  .size NhapMang, .-NhapMang  .section .rodata  .align 8  .LC2:  .string "Tong cac so le trong mang la: "  .text  .globl In  .type In, @function  In:  .LFB1:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 6, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 6  subq $32, %rsp  movq %rdi, -24(%rbp)  movl %esi, -28(%rbp)  movl $0, -4(%rbp)  leaq .LC2(%rip), %rdi  call puts@PLT  movl $0, -8(%rbp)  jmp .L5  .L6:  movl -8(%rbp), %eax  cltq  leaq 0(,%rax,4), %rdx  movq -24(%rbp), %rax  addq %rdx, %rax  movl (%rax), %eax  addl %eax, -4(%rbp)  addl $1, -8(%rbp)  .L5:  movl -8(%rbp), %eax  cmpl -28(%rbp), %eax  jl .L6  movl -4(%rbp), %eax  movl %eax, %esi  leaq .LC1(%rip), %rdi  movl $0, %eax  call printf@PLT  nop  leave  .cfi\_def\_cfa 7, 8  ret  .cfi\_endproc  .LFE1:  .size In, .-In  .section .rodata  .LC3:  .string "Nhap n: "  .text  .globl main  .type main, @function  main:  .LFB2:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 6, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 6  pushq %rbx  subq $40, %rsp  .cfi\_offset 3, -24  movq %fs:40, %rax  movq %rax, -24(%rbp)  xorl %eax, %eax  movq %rsp, %rax  movq %rax, %rbx  movl -44(%rbp), %eax  movslq %eax, %rdx  subq $1, %rdx  movq %rdx, -40(%rbp)  movslq %eax, %rdx  movq %rdx, %r8  movl $0, %r9d  movslq %eax, %rdx  movq %rdx, %rsi  movl $0, %edi  cltq  salq $2, %rax  leaq 3(%rax), %rdx  movl $16, %eax  subq $1, %rax  addq %rdx, %rax  movl $16, %ecx  movl $0, %edx  divq %rcx  imulq $16, %rax, %rax  subq %rax, %rsp  movq %rsp, %rax  addq $3, %rax  shrq $2, %rax  salq $2, %rax  movq %rax, -32(%rbp)  leaq .LC3(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq -44(%rbp), %rax  movq %rax, %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl -44(%rbp), %edx  movq -32(%rbp), %rax  movl %edx, %esi  movq %rax, %rdi  call NhapMang  movl -44(%rbp), %edx  movq -32(%rbp), %rax  movl %edx, %esi  movq %rax, %rdi  call In  movq %rbx, %rsp  movl $0, %eax  movq -24(%rbp), %rcx  xorq %fs:40, %rcx  je .L9  call \_\_stack\_chk\_fail@PLT  .L9:  movq -8(%rbp), %rbx  leave  .cfi\_def\_cfa 7, 8  ret  .cfi\_endproc  .LFE2:  .size main, .-main  .ident "GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",@progbits |

## Bai16\_8086\_TU.s

|  |
| --- |
| .file "b6.c"  .text  .section .rodata  .LC0:  .string "Nhap a[%d]: "  .LC1:  .string "%d"  .text  .globl NhapMang  .type NhapMang, @function  NhapMang:  .LFB0:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 6, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 6  subq $32, %rsp  movq %rdi, -24(%rbp)  movl %esi, -28(%rbp)  movl $0, -4(%rbp)  jmp .L2  .L3:  movl -4(%rbp), %eax  movl %eax, %esi  leaq .LC0(%rip), %rdi  movl $0, %eax  call printf@PLT  movl -4(%rbp), %eax  cltq  leaq 0(,%rax,4), %rdx  movq -24(%rbp), %rax  addq %rdx, %rax  movq %rax, %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  addl $1, -4(%rbp)  .L2:  movl -4(%rbp), %eax  cmpl -28(%rbp), %eax  jl .L3  nop  leave  .cfi\_def\_cfa 2, 8  ret  .cfi\_endproc  .LFE0:  .size NhapMang, .-NhapMang  .section .rodata  .align 8  .LC2:  .string "Tong cac so le trong mang la: "  .text  .globl In  .type In, @function  In:  .LFB1:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 1, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 1  subq $32, %rsp  movq %rdi, -24(%rbp)  movl %esi, -28(%rbp)  movl $0, -4(%rbp)  leaq .LC2(%rip), %rdi  call puts@PLT  movl $0, -8(%rbp)  jmp .L5  .L6:  movl -8(%rbp), %eax  cltq  leaq 0(,%rax,4), %rdx  movq -24(%rbp), %rax  addq %rdx, %rax  movl (%rax), %eax  addl %eax, -4(%rbp)  addl $1, -8(%rbp)  .L5:  movl -8(%rbp), %eax  cmpl -28(%rbp), %eax  jl .L6  movl -4(%rbp), %eax  movl %eax, %esi  leaq .LC1(%rip), %rdi  movl $0, %eax  call printf@PLT  nop  leave  .cfi\_def\_cfa 2, 8  ret  .cfi\_endproc  .LFE1:  .size In, .-In  .section .rodata  .LC3:  .string "Nhap n: "  .text  .globl main  .type main, @function  main:  .LFB2:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 1, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 1  pushq %rbx  subq $40, %rsp  .cfi\_offset 3, -24  movq %fs:40, %rax  movq %rax, -24(%rbp)  xorl %eax, %eax  movq %rsp, %rax  movq %rax, %rbx  movl -44(%rbp), %eax  movslq %eax, %rdx  subq $1, %rdx  movq %rdx, -40(%rbp)  movslq %eax, %rdx  movq %rdx, %r8  movl $0, %r9d  movslq %eax, %rdx  movq %rdx, %rsi  movl $0, %edi  cltq  salq $2, %rax  leaq 3(%rax), %rdx  movl $16, %eax  subq $1, %rax  addq %rdx, %rax  movl $16, %ecx  movl $0, %edx  divq %rcx  imulq $16, %rax, %rax  subq %rax, %rsp  movq %rsp, %rax  addq $3, %rax  shrq $2, %rax  salq $2, %rax  movq %rax, -32(%rbp)  leaq .LC3(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq -44(%rbp), %rax  movq %rax, %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl -44(%rbp), %edx  movq -32(%rbp), %rax  movl %edx, %esi  movq %rax, %rdi  call NhapMang  movl -44(%rbp), %edx  movq -32(%rbp), %rax  movl %edx, %esi  movq %rax, %rdi  call In  movq %rbx, %rsp  movl $0, %eax  movq -24(%rbp), %rcx  xorq %fs:40, %rcx  je .L9  call \_\_stack\_chk\_fail@PLT  .L9:  movq -8(%rbp), %rbx  leave  .cfi\_def\_cfa 2, 8  ret  .cfi\_endproc  .LFE2:  .size main, .-main  .ident "GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",@progbits |

## Bai16\_ARM.s

|  |
| --- |
| .arch armv5t  .eabi\_attribute 20, 1  .eabi\_attribute 21, 1  .eabi\_attribute 23, 3  .eabi\_attribute 24, 1  .eabi\_attribute 25, 1  .eabi\_attribute 26, 2  .eabi\_attribute 30, 6  .eabi\_attribute 34, 0  .eabi\_attribute 18, 4  .file "b6.c"  .text  .section .rodata  .align 2  .LC0:  .ascii "Nhap a[%d]: \000"  .align 2  .LC1:  .ascii "%d\000"  .text  .align 2  .global NhapMang  .syntax unified  .arm  .fpu softvfp  .type NhapMang, %function  NhapMang:  @ args = 0, pretend = 0, frame = 16  @ frame\_needed = 1, uses\_anonymous\_args = 0  push {fp, lr}  add fp, sp, #4  sub sp, sp, #16  str r0, [fp, #-16]  str r1, [fp, #-20]  mov r3, #0  str r3, [fp, #-8]  b .L2  .L3:  ldr r1, [fp, #-8]  ldr r0, .L4  bl printf  ldr r3, [fp, #-8]  lsl r3, r3, #2  ldr r2, [fp, #-16]  add r3, r2, r3  mov r1, r3  ldr r0, .L4+4  bl \_\_isoc99\_scanf  ldr r3, [fp, #-8]  add r3, r3, #1  str r3, [fp, #-8]  .L2:  ldr r2, [fp, #-8]  ldr r3, [fp, #-20]  cmp r2, r3  blt .L3  nop  sub sp, fp, #4  @ sp needed  pop {fp, pc}  .L5:  .align 2  .L4:  .word .LC0  .word .LC1  .size NhapMang, .-NhapMang  .section .rodata  .align 2  .LC2:  .ascii "Tong cac so le trong mang la: \000"  .text  .align 2  .global In  .syntax unified  .arm  .fpu softvfp  .type In, %function  In:  @ args = 0, pretend = 0, frame = 16  @ frame\_needed = 1, uses\_anonymous\_args = 0  push {fp, lr}  add fp, sp, #4  sub sp, sp, #16  str r0, [fp, #-16]  str r1, [fp, #-20]  mov r3, #0  str r3, [fp, #-8]  ldr r0, .L9  bl puts  mov r3, #0  str r3, [fp, #-12]  b .L7  .L8:  ldr r3, [fp, #-12]  lsl r3, r3, #2  ldr r2, [fp, #-16]  add r3, r2, r3  ldr r3, [r3]  ldr r2, [fp, #-8]  add r3, r2, r3  str r3, [fp, #-8]  ldr r3, [fp, #-12]  add r3, r3, #1  str r3, [fp, #-12]  .L7:  ldr r2, [fp, #-12]  ldr r3, [fp, #-20]  cmp r2, r3  blt .L8  ldr r1, [fp, #-8]  ldr r0, .L9+4  bl printf  nop  sub sp, fp, #4  @ sp needed  pop {fp, pc}  .L10:  .align 2  .L9:  .word .LC2  .word .LC1  .size In, .-In  .section .rodata  .align 2  .LC3:  .ascii "Nhap n: \000"  .text  .align 2  .global main  .syntax unified  .arm  .fpu softvfp  .type main, %function  main:  @ args = 0, pretend = 0, frame = 16  @ frame\_needed = 1, uses\_anonymous\_args = 0  push {r4, r5, fp, lr}  add fp, sp, #12  sub sp, sp, #16  ldr r3, .L14  ldr r3, [r3]  str r3, [fp, #-16]  mov r3, sp  mov r5, r3  ldr r0, [fp, #-28]  sub r3, r0, #1  str r3, [fp, #-24]  mov r3, r0  mov r1, r3  mov r2, #0  mov r3, #0  mov r4, #0  lsl r4, r2, #5  orr r4, r4, r1, lsr #27  lsl r3, r1, #5  mov r3, r0  mov r1, r3  mov r2, #0  mov r3, #0  mov r4, #0  lsl r4, r2, #5  orr r4, r4, r1, lsr #27  lsl r3, r1, #5  mov r3, r0  lsl r3, r3, #2  add r3, r3, #3  add r3, r3, #7  lsr r3, r3, #3  lsl r3, r3, #3  sub sp, sp, r3  mov r3, sp  add r3, r3, #3  lsr r3, r3, #2  lsl r3, r3, #2  str r3, [fp, #-20]  ldr r0, .L14+4  bl printf  sub r3, fp, #28  mov r1, r3  ldr r0, .L14+8  bl \_\_isoc99\_scanf  ldr r3, [fp, #-20]  ldr r2, [fp, #-28]  mov r1, r2  mov r0, r3  bl NhapMang  ldr r3, [fp, #-20]  ldr r2, [fp, #-28]  mov r1, r2  mov r0, r3  bl In  mov sp, r5  mov r3, #0  mov r0, r3  ldr r3, .L14  ldr r2, [fp, #-16]  ldr r3, [r3]  cmp r2, r3  beq .L13  bl \_\_stack\_chk\_fail  .L13:  sub sp, fp, #12  @ sp needed  pop {r4, r5, fp, pc}  .L15:  .align 2  .L14:  .word \_\_stack\_chk\_guard  .word .LC3  .word .LC1  .size main, .-main  .ident "GCC: (Ubuntu/Linaro 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",%progbits |

## Bai16\_ARM\_TU.s

|  |
| --- |
| .arch armv5t  .eabi\_attribute 20, 1  .eabi\_attribute 21, 1  .eabi\_attribute 23, 3  .eabi\_attribute 24, 1  .eabi\_attribute 25, 1  .eabi\_attribute 26, 2  .eabi\_attribute 30, 2  .eabi\_attribute 34, 0  .eabi\_attribute 18, 4  .file "b6.c"  .text  .section .rodata.str1.4,"aMS",%progbits,1  .align 2  .LC0:  .ascii "Nhap a[%d]: \000"  .align 2  .LC1:  .ascii "%d\000"  .text  .align 2  .global NhapMang  .syntax unified  .arm  .fpu softvfp  .type NhapMang, %function  NhapMang:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 0, uses\_anonymous\_args = 0  push {r4, r5, r6, r7, r8, lr}  subs r6, r1, #0  pople {r4, r5, r6, r7, r8, pc}  mov r5, r0  mov r4, #0  ldr r8, .L7  ldr r7, .L7+4  .L3:  mov r0, #1  mov r2, r4  mov r1, r8  add r4, r4, r0  bl \_\_printf\_chk  mov r1, r5  mov r0, r7  bl \_\_isoc99\_scanf  cmp r6, r4  add r5, r5, #4  bne .L3  pop {r4, r5, r6, r7, r8, pc}  .L8:  .align 2  .L7:  .word .LC0  .word .LC1  .size NhapMang, .-NhapMang  .section .rodata.str1.4  .align 2  .LC2:  .ascii "Tong cac so le trong mang la: \000"  .text  .align 2  .global In  .syntax unified  .arm  .fpu softvfp  .type In, %function  In:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 0, uses\_anonymous\_args = 0  push {r4, r5, r6, lr}  mov r5, r1  mov r4, r0  ldr r0, .L15  bl puts  cmp r5, #0  mov r2, #0  ble .L10  sub r0, r4, #4  add r1, r0, r5, lsl #2  .L11:  ldr r3, [r0, #4]!  cmp r0, r1  add r2, r2, r3  bne .L11  .L10:  pop {r4, r5, r6, lr}  mov r0, #1  ldr r1, .L15+4  b \_\_printf\_chk  .L16:  .align 2  .L15:  .word .LC2  .word .LC1  .size In, .-In  .section .rodata.str1.4  .align 2  .LC3:  .ascii "Nhap n: \000"  .section .rodata.cst4,"aM",%progbits,4  .align 2  .LC4:  .word \_\_stack\_chk\_guard  .section .text.startup,"ax",%progbits  .align 2  .global main  .syntax unified  .arm  .fpu softvfp  .type main, %function  main:  @ args = 0, pretend = 0, frame = 8  @ frame\_needed = 1, uses\_anonymous\_args = 0  push {r4, fp, lr}  add fp, sp, #8  sub sp, sp, #12  ldr r3, [fp, #-20]  ldr r1, .L21  lsl r3, r3, #2  add r3, r3, #7  bic r3, r3, #7  sub sp, sp, r3  mov r4, sp  ldr r3, .L21+4  mov r0, #1  ldr r3, [r3]  str r3, [fp, #-16]  mov r3,#0  bl \_\_printf\_chk  sub r1, fp, #20  ldr r0, .L21+8  bl \_\_isoc99\_scanf  mov r0, r4  ldr r1, [fp, #-20]  bl NhapMang  mov r0, r4  ldr r1, [fp, #-20]  bl In  ldr r3, .L21+4  ldr r2, [r3]  ldr r3, [fp, #-16]  eors r2, r3, r2  mov r3, #0  bne .L20  mov r0, #0  sub sp, fp, #8  @ sp needed  pop {r4, fp, pc}  .L20:  bl \_\_stack\_chk\_fail  .L22:  .align 2  .L21:  .word .LC3  .word .LC4  .word .LC1  .size main, .-main  .ident "GCC: (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0"  .section .note.GNU-stack,"",%progbits |

# Vd17:

## Bai17.c

|  |
| --- |
| #include<stdio.h>  int a[50];  int i,n,j,kt;  main()  {  printf("Nhap so luong phan tu:");  scanf("%d",&n);  for(i=0;i<n;i++)  {  scanf("%d",&a[i]);  }  printf("\nCac so nguyen to co trong mang va vi tri cac so do trong mang la:");  for(i=0;i<n;i++)  {  kt=0;  for(j=2;j<=a[i]/2;j++)  {  if(a[i]%j==0)  kt=1;  }  if(kt==0)  printf("\nso nguyen to %d vi tri %d trong mang ", a[i], i );  }    } |

## Bai17\_TU.c

|  |
| --- |
| #include<stdio.h>  void main()  {  int a[50];  int i,n,j,kt;  printf("Nhap so luong phan tu:");  scanf("%d",&n);  for(i=0;i<n;i++)  {  scanf("%d",&a[i]);  }  printf("\nCac so nguyen to co trong mang va vi tri cac so do trong mang la:");  for(i=0;i<n;i++)  {  kt=0;  for(j=2;j<=a[i]/2;j++)  {  if(a[i]%j==0)  kt=1;  }  if(kt==0)  printf("\nso nguyen to %d vi tri %d trong mang ", a[i], i );  }  } |

## Bai17\_8086.s

|  |
| --- |
| .file "b7.c"  .text  .comm a,200,32  .comm i,4,4  .comm n,4,4  .comm j,4,4  .comm kt,4,4  .section .rodata  .LC0:  .string "Nhap so luong phan tu:"  .LC1:  .string "%d"  .align 8  .LC2:  .string "\nCac so nguyen to co trong mang va vi tri cac so do trong mang la:"  .align 8  .LC3:  .string "\nso nguyen to %d vi tri %d trong mang "  .text  .globl main  .type main, @function  main:  .LFB0:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 6, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 6  leaq .LC0(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq n(%rip), %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl $0, i(%rip)  jmp .L2  .L3:  movl i(%rip), %eax  cltq  leaq 0(,%rax,4), %rdx  leaq a(%rip), %rax  addq %rdx, %rax  movq %rax, %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl i(%rip), %eax  addl $1, %eax  movl %eax, i(%rip)  .L2:  movl i(%rip), %edx  movl n(%rip), %eax  cmpl %eax, %edx  jl .L3  leaq .LC2(%rip), %rdi  movl $0, %eax  call printf@PLT  movl $0, i(%rip)  jmp .L4  .L9:  movl $0, kt(%rip)  movl $2, j(%rip)  jmp .L5  .L7:  movl i(%rip), %eax  cltq  leaq 0(,%rax,4), %rdx  leaq a(%rip), %rax  movl (%rdx,%rax), %eax  movl j(%rip), %ecx  cltd  idivl %ecx  movl %edx, %eax  testl %eax, %eax  jne .L6  movl $1, kt(%rip)  .L6:  movl j(%rip), %eax  addl $1, %eax  movl %eax, j(%rip)  .L5:  movl i(%rip), %eax  cltq  leaq 0(,%rax,4), %rdx  leaq a(%rip), %rax  movl (%rdx,%rax), %eax  movl %eax, %edx  shrl $31, %edx  addl %edx, %eax  sarl %eax  movl %eax, %edx  movl j(%rip), %eax  cmpl %eax, %edx  jge .L7  movl kt(%rip), %eax  testl %eax, %eax  jne .L8  movl i(%rip), %edx  movl i(%rip), %eax  cltq  leaq 0(,%rax,4), %rcx  leaq a(%rip), %rax  movl (%rcx,%rax), %eax  movl %eax, %esi  leaq .LC3(%rip), %rdi  movl $0, %eax  call printf@PLT  .L8:  movl i(%rip), %eax  addl $1, %eax  movl %eax, i(%rip)  .L4:  movl i(%rip), %edx  movl n(%rip), %eax  cmpl %eax, %edx  jl .L9  movl $0, %eax  popq %rbp  .cfi\_def\_cfa 7, 8  ret  .cfi\_endproc  .LFE0:  .size main, .-main  .ident "GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",@progbits |

## Bai17\_8086\_TU.s

|  |
| --- |
| .file "b7.c"  .text  .comm a,200,32  .comm i,4,4  .comm n,4,4  .comm j,4,4  .comm kt,4,4  .section .rodata  .LC0:  .string "Nhap so luong phan tu:"  .LC1:  .string "%d"  .align 8  .LC2:  .string "\nCac so nguyen to co trong mang va vi tri cac so do trong mang la:"  .align 8  .LC3:  .string "\nso nguyen to %d vi tri %d trong mang "  .text  .globl main  .type main, @function  main:  .LFB0:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 1, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 1  leaq .LC0(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq n(%rip), %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl $0, i(%rip)  jmp .L2  .L3:  movl i(%rip), %eax  cltq  leaq 0(,%rax,4), %rdx  leaq a(%rip), %rax  addq %rdx, %rax  movq %rax, %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl i(%rip), %eax  addl $1, %eax  movl %eax, i(%rip)  .L2:  movl i(%rip), %edx  movl n(%rip), %eax  cmpl %eax, %edx  jl .L3  leaq .LC2(%rip), %rdi  movl $0, %eax  call printf@PLT  movl $0, i(%rip)  jmp .L4  .L9:  movl $0, kt(%rip)  movl $2, j(%rip)  jmp .L5  .L7:  movl i(%rip), %eax  cltq  leaq 0(,%rax,4), %rdx  leaq a(%rip), %rax  movl (%rdx,%rax), %eax  movl j(%rip), %ecx  cltd  idivl %ecx  movl %edx, %eax  testl %eax, %eax  jne .L6  movl $1, kt(%rip)  .L6:  movl j(%rip), %eax  addl $1, %eax  movl %eax, j(%rip)  .L5:  movl i(%rip), %eax  cltq  leaq 0(,%rax,4), %rdx  leaq a(%rip), %rax  movl (%rdx,%rax), %eax  movl %eax, %edx  shrl $31, %edx  addl %edx, %eax  sarl %eax  movl %eax, %edx  movl j(%rip), %eax  cmpl %eax, %edx  jge .L7  movl kt(%rip), %eax  testl %eax, %eax  jne .L8  movl i(%rip), %edx  movl i(%rip), %eax  cltq  leaq 0(,%rax,4), %rcx  leaq a(%rip), %rax  movl (%rcx,%rax), %eax  movl %eax, %esi  leaq .LC3(%rip), %rdi  movl $0, %eax  call printf@PLT  .L8:  movl i(%rip), %eax  addl $1, %eax  movl %eax, i(%rip)  .L4:  movl i(%rip), %edx  movl n(%rip), %eax  cmpl %eax, %edx  jl .L9  movl $0, %eax  popq %rbp  .cfi\_def\_cfa 2, 8  ret  .cfi\_endproc  .LFE0:  .size main, .-main  .ident "GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",@progbits |

## Bai17\_ARM.s

|  |
| --- |
| .arch armv5t  .eabi\_attribute 20, 1  .eabi\_attribute 21, 1  .eabi\_attribute 23, 3  .eabi\_attribute 24, 1  .eabi\_attribute 25, 1  .eabi\_attribute 26, 2  .eabi\_attribute 30, 6  .eabi\_attribute 34, 0  .eabi\_attribute 18, 4  .file "b7.c"  .text  .comm a,200,4  .comm i,4,4  .comm n,4,4  .comm j,4,4  .comm kt,4,4  .section .rodata  .align 2  .LC0:  .ascii "Nhap so luong phan tu:\000"  .align 2  .LC1:  .ascii "%d\000"  .align 2  .LC2:  .ascii "\012Cac so nguyen to co trong mang va vi tri cac s"  .ascii "o do trong mang la:\000"  .global \_\_aeabi\_idivmod  .align 2  .LC3:  .ascii "\012so nguyen to %d vi tri %d trong mang \000"  .text  .align 2  .global main  .syntax unified  .arm  .fpu softvfp  .type main, %function  main:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 1, uses\_anonymous\_args = 0  push {fp, lr}  add fp, sp, #4  ldr r0, .L11  bl printf  ldr r1, .L11+4  ldr r0, .L11+8  bl \_\_isoc99\_scanf  ldr r3, .L11+12  mov r2, #0  str r2, [r3]  b .L2  .L3:  ldr r3, .L11+12  ldr r3, [r3]  lsl r3, r3, #2  ldr r2, .L11+16  add r3, r3, r2  mov r1, r3  ldr r0, .L11+8  bl \_\_isoc99\_scanf  ldr r3, .L11+12  ldr r3, [r3]  add r3, r3, #1  ldr r2, .L11+12  str r3, [r2]  .L2:  ldr r3, .L11+12  ldr r2, [r3]  ldr r3, .L11+4  ldr r3, [r3]  cmp r2, r3  blt .L3  ldr r0, .L11+20  bl printf  ldr r3, .L11+12  mov r2, #0  str r2, [r3]  b .L4  .L9:  ldr r3, .L11+24  mov r2, #0  str r2, [r3]  ldr r3, .L11+28  mov r2, #2  str r2, [r3]  b .L5  .L7:  ldr r3, .L11+12  ldr r3, [r3]  ldr r2, .L11+16  ldr r2, [r2, r3, lsl #2]  ldr r3, .L11+28  ldr r3, [r3]  mov r1, r3  mov r0, r2  bl \_\_aeabi\_idivmod  mov r3, r1  cmp r3, #0  bne .L6  ldr r3, .L11+24  mov r2, #1  str r2, [r3]  .L6:  ldr r3, .L11+28  ldr r3, [r3]  add r3, r3, #1  ldr r2, .L11+28  str r3, [r2]  .L5:  ldr r3, .L11+12  ldr r3, [r3]  ldr r2, .L11+16  ldr r3, [r2, r3, lsl #2]  lsr r2, r3, #31  add r3, r2, r3  asr r3, r3, #1  mov r2, r3  ldr r3, .L11+28  ldr r3, [r3]  cmp r2, r3  bge .L7  ldr r3, .L11+24  ldr r3, [r3]  cmp r3, #0  bne .L8  ldr r3, .L11+12  ldr r3, [r3]  ldr r2, .L11+16  ldr r1, [r2, r3, lsl #2]  ldr r3, .L11+12  ldr r3, [r3]  mov r2, r3  ldr r0, .L11+32  bl printf  .L8:  ldr r3, .L11+12  ldr r3, [r3]  add r3, r3, #1  ldr r2, .L11+12  str r3, [r2]  .L4:  ldr r3, .L11+12  ldr r2, [r3]  ldr r3, .L11+4  ldr r3, [r3]  cmp r2, r3  blt .L9  mov r3, #0  mov r0, r3  pop {fp, pc}  .L12:  .align 2  .L11:  .word .LC0  .word n  .word .LC1  .word i  .word a  .word .LC2  .word kt  .word j  .word .LC3  .size main, .-main  .ident "GCC: (Ubuntu/Linaro 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",%progbits |

## Bai17\_ARM\_TU.s

|  |
| --- |
| .arch armv5t  .eabi\_attribute 20, 1  .eabi\_attribute 21, 1  .eabi\_attribute 23, 3  .eabi\_attribute 24, 1  .eabi\_attribute 25, 1  .eabi\_attribute 26, 2  .eabi\_attribute 30, 2  .eabi\_attribute 34, 0  .eabi\_attribute 18, 4  .file "b7.c"  .text  .section .rodata.str1.4,"aMS",%progbits,1  .align 2  .LC0:  .ascii "Nhap so luong phan tu:\000"  .align 2  .LC1:  .ascii "%d\000"  .align 2  .LC2:  .ascii "\012Cac so nguyen to co trong mang va vi tri cac s"  .ascii "o do trong mang la:\000"  .align 2  .LC3:  .ascii "\012so nguyen to %d vi tri %d trong mang \000"  .global \_\_aeabi\_idivmod  .section .text.startup,"ax",%progbits  .align 2  .global main  .syntax unified  .arm  .fpu softvfp  .type main, %function  main:  @ args = 0, pretend = 0, frame = 8  @ frame\_needed = 0, uses\_anonymous\_args = 0  push {r4, r5, r6, r7, r8, r9, r10, fp, lr}  ldr r9, .L25  ldr r1, .L25+4  sub sp, sp, #12  mov r0, #1  bl \_\_printf\_chk  mov r1, r9  ldr r0, .L25+8  bl \_\_isoc99\_scanf  mov r1, #0  ldr r3, [r9]  ldr r8, .L25+12  cmp r3, r1  str r1, [r8]  ble .L5  ldr r5, .L25+16  ldr r4, .L25+8  .L2:  add r1, r5, r1, lsl #2  mov r0, r4  bl \_\_isoc99\_scanf  ldr r1, [r8]  ldr r3, [r9]  add r1, r1, #1  cmp r1, r3  str r1, [r8]  blt .L2  .L5:  mov r6, #0  mov r0, #1  ldr r1, .L25+20  bl \_\_printf\_chk  ldr r3, [r9]  str r6, [r8]  cmp r3, r6  ldrgt r3, .L25+16  strgt r3, [sp, #4]  ble .L22  .L3:  mov r2, #0  ldr r3, [sp, #4]  ldr fp, [r3, r6, lsl #2]  ldr r3, .L25+24  cmp fp, #3  str r2, [r3]  movgt r7, #0  mov r2, #2  ldr r3, .L25+28  add r5, fp, fp, lsr #31  movgt r4, #2  movgt r10, r7  str r2, [r3]  asr r5, r5, #1  ble .L9  .L6:  mov r1, r4  mov r0, fp  bl \_\_aeabi\_idivmod  cmp r1, #0  moveq r7, #1  add r4, r4, #1  moveq r10, r7  cmp r4, r5  ble .L6  cmp fp, #3  addgt r5, r5, #1  movle r5, #3  ldr r3, .L25+28  cmp r7, #0  str r5, [r3]  beq .L9  ldr r3, .L25+24  cmp r10, #0  str r10, [r3]  beq .L9  .L7:  ldr r6, [r8]  ldr r3, [r9]  add r6, r6, #1  cmp r6, r3  str r6, [r8]  blt .L3  .L22:  mov r0, #0  add sp, sp, #12  @ sp needed  pop {r4, r5, r6, r7, r8, r9, r10, fp, pc}  .L9:  mov r3, r6  mov r2, fp  ldr r1, .L25+32  mov r0, #1  bl \_\_printf\_chk  b .L7  .L26:  .align 2  .L25:  .word n  .word .LC0  .word .LC1  .word i  .word a  .word .LC2  .word kt  .word j  .word .LC3  .size main, .-main  .comm kt,4,4  .comm j,4,4  .comm n,4,4  .comm i,4,4  .comm a,200,4  .ident "GCC: (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0"  .section .note.GNU-stack,"",%progbits |

# Vd18:

## Bai18.c

|  |
| --- |
| #include<stdio.h>  int a[50];  int i,n,tg,max,j,s=0;  main()  {    printf("nhap vao so phan tu: ");  scanf("%d",&n);  for(i=0;i<n;i++)  {  scanf("%d", &a[i]);  }  max=a[0];  for(i=1;i<n;i++)  {  if(a[i]>max)  {  max =a[i];  }  }  printf("\nSo lon nhat =%d",max);  printf("\nvi tri cua gia tri lon nhat trong day la: ");  for(i=0;i<n;i++)  {  if (a[i]==max)  {  printf("%6d", i+1);  }  }  } |

## Bai18\_TU.c

|  |
| --- |
| #include<stdio.h>  void main()  {  int a[50];  int i,n,tg,max,j,s=0;  printf("nhap vao so phan tu: ");  scanf("%d",&n);  for(i=0;i<n;i++)  {  scanf("%d", &a[i]);  }  max=a[0];  for(i=1;i<n;i++)  {  if(a[i]>max)  {  max =a[i];  }  }  printf("\nSo lon nhat =%d",max);  printf("\nvi tri cua gia tri lon nhat trong day la: ");  for(i=0;i<n;i++)  {  if (a[i]==max)  {  printf("%6d", i+1);  }  }  } |

## Bai18\_8086.s

|  |
| --- |
| .file "b8.c"  .text  .comm a,200,32  .comm i,4,4  .comm n,4,4  .comm tg,4,4  .comm max,4,4  .comm j,4,4  .globl s  .bss  .align 4  .type s, @object  .size s, 4  s:  .zero 4  .section .rodata  .LC0:  .string "nhap vao so phan tu: "  .LC1:  .string "%d"  .LC2:  .string "\nSo lon nhat =%d"  .align 8  .LC3:  .string "\nvi tri cua gia tri lon nhat trong day la: "  .LC4:  .string "%6d"  .text  .globl main  .type main, @function  main:  .LFB0:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 6, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 6  leaq .LC0(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq n(%rip), %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl $0, i(%rip)  jmp .L2  .L3:  movl i(%rip), %eax  cltq  leaq 0(,%rax,4), %rdx  leaq a(%rip), %rax  addq %rdx, %rax  movq %rax, %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl i(%rip), %eax  addl $1, %eax  movl %eax, i(%rip)  .L2:  movl i(%rip), %edx  movl n(%rip), %eax  cmpl %eax, %edx  jl .L3  movl a(%rip), %eax  movl %eax, max(%rip)  movl $1, i(%rip)  jmp .L4  .L6:  movl i(%rip), %eax  cltq  leaq 0(,%rax,4), %rdx  leaq a(%rip), %rax  movl (%rdx,%rax), %edx  movl max(%rip), %eax  cmpl %eax, %edx  jle .L5  movl i(%rip), %eax  cltq  leaq 0(,%rax,4), %rdx  leaq a(%rip), %rax  movl (%rdx,%rax), %eax  movl %eax, max(%rip)  .L5:  movl i(%rip), %eax  addl $1, %eax  movl %eax, i(%rip)  .L4:  movl i(%rip), %edx  movl n(%rip), %eax  cmpl %eax, %edx  jl .L6  movl max(%rip), %eax  movl %eax, %esi  leaq .LC2(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq .LC3(%rip), %rdi  movl $0, %eax  call printf@PLT  movl $0, i(%rip)  jmp .L7  .L9:  movl i(%rip), %eax  cltq  leaq 0(,%rax,4), %rdx  leaq a(%rip), %rax  movl (%rdx,%rax), %edx  movl max(%rip), %eax  cmpl %eax, %edx  jne .L8  movl i(%rip), %eax  addl $1, %eax  movl %eax, %esi  leaq .LC4(%rip), %rdi  movl $0, %eax  call printf@PLT  .L8:  movl i(%rip), %eax  addl $1, %eax  movl %eax, i(%rip)  .L7:  movl i(%rip), %edx  movl n(%rip), %eax  cmpl %eax, %edx  jl .L9  movl $0, %eax  popq %rbp  .cfi\_def\_cfa 7, 8  ret  .cfi\_endproc  .LFE0:  .size main, .-main  .ident "GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",@progbits |

## Bai18\_8086\_TU.s

|  |
| --- |
| .file "b8.c"  .text  .comm a,200,32  .comm i,4,4  .comm n,4,4  .comm tg,4,4  .comm max,4,4  .comm j,4,4  .globl s  .bss  .align 4  .type s, @object  .size s, 4  s:  .zero 4  .section .rodata  .LC0:  .string "nhap vao so phan tu: "  .LC1:  .string "%d"  .LC2:  .string "\nSo lon nhat =%d"  .align 8  .LC3:  .string "\nvi tri cua gia tri lon nhat trong day la: "  .LC4:  .string "%6d"  .text  .globl main  .type main, @function  main:  .LFB0:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 1, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 1  leaq .LC0(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq n(%rip), %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl $0, i(%rip)  jmp .L2  .L3:  movl i(%rip), %eax  cltq  leaq 0(,%rax,4), %rdx  leaq a(%rip), %rax  addq %rdx, %rax  movq %rax, %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl i(%rip), %eax  addl $1, %eax  movl %eax, i(%rip)  .L2:  movl i(%rip), %edx  movl n(%rip), %eax  cmpl %eax, %edx  jl .L3  movl a(%rip), %eax  movl %eax, max(%rip)  movl $1, i(%rip)  jmp .L4  .L6:  movl i(%rip), %eax  cltq  leaq 0(,%rax,4), %rdx  leaq a(%rip), %rax  movl (%rdx,%rax), %edx  movl max(%rip), %eax  cmpl %eax, %edx  jle .L5  movl i(%rip), %eax  cltq  leaq 0(,%rax,4), %rdx  leaq a(%rip), %rax  movl (%rdx,%rax), %eax  movl %eax, max(%rip)  .L5:  movl i(%rip), %eax  addl $1, %eax  movl %eax, i(%rip)  .L4:  movl i(%rip), %edx  movl n(%rip), %eax  cmpl %eax, %edx  jl .L6  movl max(%rip), %eax  movl %eax, %esi  leaq .LC2(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq .LC3(%rip), %rdi  movl $0, %eax  call printf@PLT  movl $0, i(%rip)  jmp .L7  .L9:  movl i(%rip), %eax  cltq  leaq 0(,%rax,4), %rdx  leaq a(%rip), %rax  movl (%rdx,%rax), %edx  movl max(%rip), %eax  cmpl %eax, %edx  jne .L8  movl i(%rip), %eax  addl $1, %eax  movl %eax, %esi  leaq .LC4(%rip), %rdi  movl $0, %eax  call printf@PLT  .L8:  movl i(%rip), %eax  addl $1, %eax  movl %eax, i(%rip)  .L7:  movl i(%rip), %edx  movl n(%rip), %eax  cmpl %eax, %edx  jl .L9  movl $0, %eax  popq %rbp  .cfi\_def\_cfa 2, 8  ret  .cfi\_endproc  .LFE0:  .size main, .-main  .ident "GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",@progbits |

## Bai18\_ARM.s

|  |
| --- |
| .arch armv5t  .eabi\_attribute 20, 1  .eabi\_attribute 21, 1  .eabi\_attribute 23, 3  .eabi\_attribute 24, 1  .eabi\_attribute 25, 1  .eabi\_attribute 26, 2  .eabi\_attribute 30, 6  .eabi\_attribute 34, 0  .eabi\_attribute 18, 4  .file "b8.c"  .text  .comm a,200,4  .comm i,4,4  .comm n,4,4  .comm tg,4,4  .comm max,4,4  .comm j,4,4  .global s  .bss  .align 2  .type s, %object  .size s, 4  s:  .space 4  .section .rodata  .align 2  .LC0:  .ascii "nhap vao so phan tu: \000"  .align 2  .LC1:  .ascii "%d\000"  .align 2  .LC2:  .ascii "\012So lon nhat =%d\000"  .align 2  .LC3:  .ascii "\012vi tri cua gia tri lon nhat trong day la: \000"  .align 2  .LC4:  .ascii "%6d\000"  .text  .align 2  .global main  .syntax unified  .arm  .fpu softvfp  .type main, %function  main:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 1, uses\_anonymous\_args = 0  push {fp, lr}  add fp, sp, #4  ldr r0, .L11  bl printf  ldr r1, .L11+4  ldr r0, .L11+8  bl \_\_isoc99\_scanf  ldr r3, .L11+12  mov r2, #0  str r2, [r3]  b .L2  .L3:  ldr r3, .L11+12  ldr r3, [r3]  lsl r3, r3, #2  ldr r2, .L11+16  add r3, r3, r2  mov r1, r3  ldr r0, .L11+8  bl \_\_isoc99\_scanf  ldr r3, .L11+12  ldr r3, [r3]  add r3, r3, #1  ldr r2, .L11+12  str r3, [r2]  .L2:  ldr r3, .L11+12  ldr r2, [r3]  ldr r3, .L11+4  ldr r3, [r3]  cmp r2, r3  blt .L3  ldr r3, .L11+16  ldr r3, [r3]  ldr r2, .L11+20  str r3, [r2]  ldr r3, .L11+12  mov r2, #1  str r2, [r3]  b .L4  .L6:  ldr r3, .L11+12  ldr r3, [r3]  ldr r2, .L11+16  ldr r2, [r2, r3, lsl #2]  ldr r3, .L11+20  ldr r3, [r3]  cmp r2, r3  ble .L5  ldr r3, .L11+12  ldr r3, [r3]  ldr r2, .L11+16  ldr r3, [r2, r3, lsl #2]  ldr r2, .L11+20  str r3, [r2]  .L5:  ldr r3, .L11+12  ldr r3, [r3]  add r3, r3, #1  ldr r2, .L11+12  str r3, [r2]  .L4:  ldr r3, .L11+12  ldr r2, [r3]  ldr r3, .L11+4  ldr r3, [r3]  cmp r2, r3  blt .L6  ldr r3, .L11+20  ldr r3, [r3]  mov r1, r3  ldr r0, .L11+24  bl printf  ldr r0, .L11+28  bl printf  ldr r3, .L11+12  mov r2, #0  str r2, [r3]  b .L7  .L9:  ldr r3, .L11+12  ldr r3, [r3]  ldr r2, .L11+16  ldr r2, [r2, r3, lsl #2]  ldr r3, .L11+20  ldr r3, [r3]  cmp r2, r3  bne .L8  ldr r3, .L11+12  ldr r3, [r3]  add r3, r3, #1  mov r1, r3  ldr r0, .L11+32  bl printf  .L8:  ldr r3, .L11+12  ldr r3, [r3]  add r3, r3, #1  ldr r2, .L11+12  str r3, [r2]  .L7:  ldr r3, .L11+12  ldr r2, [r3]  ldr r3, .L11+4  ldr r3, [r3]  cmp r2, r3  blt .L9  mov r3, #0  mov r0, r3  pop {fp, pc}  .L12:  .align 2  .L11:  .word .LC0  .word n  .word .LC1  .word i  .word a  .word max  .word .LC2  .word .LC3  .word .LC4  .size main, .-main  .ident "GCC: (Ubuntu/Linaro 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",%progbits |

## Bai18\_ARM\_TU.s

|  |
| --- |
| .arch armv5t  .eabi\_attribute 20, 1  .eabi\_attribute 21, 1  .eabi\_attribute 23, 3  .eabi\_attribute 24, 1  .eabi\_attribute 25, 1  .eabi\_attribute 26, 2  .eabi\_attribute 30, 2  .eabi\_attribute 34, 0  .eabi\_attribute 18, 4  .file "b8.c"  .text  .section .rodata.str1.4,"aMS",%progbits,1  .align 2  .LC0:  .ascii "nhap vao so phan tu: \000"  .align 2  .LC1:  .ascii "%d\000"  .align 2  .LC2:  .ascii "\012So lon nhat =%d\000"  .align 2  .LC3:  .ascii "\012vi tri cua gia tri lon nhat trong day la: \000"  .align 2  .LC4:  .ascii "%6d\000"  .section .text.startup,"ax",%progbits  .align 2  .global main  .syntax unified  .arm  .fpu softvfp  .type main, %function  main:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 0, uses\_anonymous\_args = 0  push {r4, r5, r6, r7, r8, lr}  ldr r5, .L23  ldr r1, .L23+4  mov r0, #1  bl \_\_printf\_chk  mov r1, r5  ldr r0, .L23+8  bl \_\_isoc99\_scanf  mov r1, #0  ldr r3, [r5]  ldr r4, .L23+12  cmp r3, r1  str r1, [r4]  ble .L2  ldr r6, .L23+16  ldr r7, .L23+8  .L3:  add r1, r6, r1, lsl #2  mov r0, r7  bl \_\_isoc99\_scanf  ldr r1, [r4]  ldr r0, [r5]  add r1, r1, #1  cmp r1, r0  str r1, [r4]  blt .L3  mov r3, #1  ldr r1, [r6]  ldr r7, .L23+20  cmp r0, r3  str r3, [r4]  str r1, [r7]  ble .L5  mov lr, #0  ldr r3, .L23+24  add ip, r6, r0, lsl #2  .L7:  ldr r2, [r3], #4  cmp r2, r1  movgt r1, r2  movgt lr, #1  cmp ip, r3  bne .L7  cmp lr, #0  str r0, [r4]  strne r1, [r7]  .L5:  ldr r2, [r7]  ldr r1, .L23+28  mov r0, #1  bl \_\_printf\_chk  mov r0, #1  ldr r1, .L23+32  bl \_\_printf\_chk  mov r2, #0  ldr r3, [r5]  str r2, [r4]  cmp r3, r2  ble .L20  ldr r8, .L23+36  b .L9  .L10:  ldr r2, [r4]  ldr r3, [r5]  add r2, r2, #1  cmp r2, r3  str r2, [r4]  bge .L20  .L9:  ldr r0, [r6, r2, lsl #2]  ldr r3, [r7]  cmp r0, r3  bne .L10  mov r1, r8  add r2, r2, #1  mov r0, #1  bl \_\_printf\_chk  b .L10  .L20:  mov r0, #0  pop {r4, r5, r6, r7, r8, pc}  .L2:  mov r2, #1  ldr r6, .L23+16  ldr r7, .L23+20  ldr r3, [r6]  str r2, [r4]  str r3, [r7]  b .L5  .L24:  .align 2  .L23:  .word n  .word .LC0  .word .LC1  .word i  .word a  .word max  .word a+4  .word .LC2  .word .LC3  .word .LC4  .size main, .-main  .global s  .comm j,4,4  .comm max,4,4  .comm tg,4,4  .comm n,4,4  .comm i,4,4  .comm a,200,4  .bss  .align 2  .type s, %object  .size s, 4  s:  .space 4  .ident "GCC: (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0"  .section .note.GNU-stack,"",%progbits |

# Vd19:

## Bai19.c

|  |
| --- |
| #include <stdio.h>    const int MAX = 100;      void NhapMang(int a[], int n){  for(int i = 0;i < n; ++i){  printf("\nNhap phan tu a[%d] = ", i);  scanf("%d", &a[i]);  }  }    void XuatMang(int a[], int n){  for(int i = 0;i < n; ++i){  printf("\nPhan tu a[%d] = %d", i, a[i]);  }  }    int TimKiem(int a[], int n, int v){  for(int i = 0;i < n; ++i){  if(a[i] == v){  return i;  }  }  return -1;  }    int main(){  int arr[MAX];  int n;  printf("\nNhap so luong phan tu: ");  do{  scanf("%d", &n);  if(n <= 0 || n > MAX){  printf("\nNhap lai so luong phan tu: ");  }  }while(n <= 0 || n > MAX);  printf("\n======NHAP MANG=====\n");  NhapMang(arr, n);  printf("\n======XUAT MANG=====\n");  XuatMang(arr, n);  } |

## Bai19\_TU.c

|  |
| --- |
| #include<stdio.h>  void main()  {  int a[50];  int i,n;  printf("Nhap so phan tu mang: ");  scanf("%d",&n);  for(i=0;i<n;i++)  {  scanf("%d",&a[i]);  }  printf("\nMang vua nhap la:");  for(i=0;i<n;i++)  {  printf("%5d",a[i]);  }    } |

## Bai19\_8086.s

|  |
| --- |
| .file "b9.c"  .text  .globl MAX  .section .rodata  .align 4  .type MAX, @object  .size MAX, 4  MAX:  .long 100  .LC0:  .string "\nNhap phan tu a[%d] = "  .LC1:  .string "%d"  .text  .globl NhapMang  .type NhapMang, @function  NhapMang:  .LFB0:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 6, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 6  subq $32, %rsp  movq %rdi, -24(%rbp)  movl %esi, -28(%rbp)  movl $0, -4(%rbp)  jmp .L2  .L3:  movl -4(%rbp), %eax  movl %eax, %esi  leaq .LC0(%rip), %rdi  movl $0, %eax  call printf@PLT  movl -4(%rbp), %eax  cltq  leaq 0(,%rax,4), %rdx  movq -24(%rbp), %rax  addq %rdx, %rax  movq %rax, %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  addl $1, -4(%rbp)  .L2:  movl -4(%rbp), %eax  cmpl -28(%rbp), %eax  jl .L3  nop  leave  .cfi\_def\_cfa 7, 8  ret  .cfi\_endproc  .LFE0:  .size NhapMang, .-NhapMang  .section .rodata  .LC2:  .string "\nPhan tu a[%d] = %d"  .text  .globl XuatMang  .type XuatMang, @function  XuatMang:  .LFB1:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 6, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 6  subq $32, %rsp  movq %rdi, -24(%rbp)  movl %esi, -28(%rbp)  movl $0, -4(%rbp)  jmp .L5  .L6:  movl -4(%rbp), %eax  cltq  leaq 0(,%rax,4), %rdx  movq -24(%rbp), %rax  addq %rdx, %rax  movl (%rax), %edx  movl -4(%rbp), %eax  movl %eax, %esi  leaq .LC2(%rip), %rdi  movl $0, %eax  call printf@PLT  addl $1, -4(%rbp)  .L5:  movl -4(%rbp), %eax  cmpl -28(%rbp), %eax  jl .L6  nop  leave  .cfi\_def\_cfa 7, 8  ret  .cfi\_endproc  .LFE1:  .size XuatMang, .-XuatMang  .globl TimKiem  .type TimKiem, @function  TimKiem:  .LFB2:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 6, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 6  movq %rdi, -24(%rbp)  movl %esi, -28(%rbp)  movl %edx, -32(%rbp)  movl $0, -4(%rbp)  jmp .L8  .L11:  movl -4(%rbp), %eax  cltq  leaq 0(,%rax,4), %rdx  movq -24(%rbp), %rax  addq %rdx, %rax  movl (%rax), %eax  cmpl %eax, -32(%rbp)  jne .L9  movl -4(%rbp), %eax  jmp .L10  .L9:  addl $1, -4(%rbp)  .L8:  movl -4(%rbp), %eax  cmpl -28(%rbp), %eax  jl .L11  movl $-1, %eax  .L10:  popq %rbp  .cfi\_def\_cfa 7, 8  ret  .cfi\_endproc  .LFE2:  .size TimKiem, .-TimKiem  .section .rodata  .LC3:  .string "\nNhap so luong phan tu: "  .LC4:  .string "\nNhap lai so luong phan tu: "  .LC5:  .string "\n======NHAP MANG====="  .LC6:  .string "\n======XUAT MANG====="  .text  .globl main  .type main, @function  main:  .LFB3:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 6, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 6  pushq %rbx  subq $40, %rsp  .cfi\_offset 3, -24  movq %fs:40, %rax  movq %rax, -24(%rbp)  xorl %eax, %eax  movq %rsp, %rax  movq %rax, %rbx  movl $100, %eax  cltq  subq $1, %rax  movq %rax, -40(%rbp)  movl $100, %eax  cltq  movq %rax, %rsi  movl $0, %edi  movl $100, %eax  cltq  movq %rax, %rdx  movl $0, %ecx  movl $100, %eax  cltq  salq $2, %rax  leaq 3(%rax), %rdx  movl $16, %eax  subq $1, %rax  addq %rdx, %rax  movl $16, %ecx  movl $0, %edx  divq %rcx  imulq $16, %rax, %rax  subq %rax, %rsp  movq %rsp, %rax  addq $3, %rax  shrq $2, %rax  salq $2, %rax  movq %rax, -32(%rbp)  leaq .LC3(%rip), %rdi  movl $0, %eax  call printf@PLT  .L15:  leaq -44(%rbp), %rax  movq %rax, %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl -44(%rbp), %eax  testl %eax, %eax  jle .L13  movl -44(%rbp), %eax  movl $100, %edx  cmpl %edx, %eax  jle .L14  .L13:  leaq .LC4(%rip), %rdi  movl $0, %eax  call printf@PLT  .L14:  movl -44(%rbp), %eax  testl %eax, %eax  jle .L15  movl -44(%rbp), %eax  movl $100, %edx  cmpl %edx, %eax  jg .L15  leaq .LC5(%rip), %rdi  call puts@PLT  movl -44(%rbp), %edx  movq -32(%rbp), %rax  movl %edx, %esi  movq %rax, %rdi  call NhapMang  leaq .LC6(%rip), %rdi  call puts@PLT  movl -44(%rbp), %edx  movq -32(%rbp), %rax  movl %edx, %esi  movq %rax, %rdi  call XuatMang  movq %rbx, %rsp  movl $0, %eax  movq -24(%rbp), %rcx  xorq %fs:40, %rcx  je .L17  call \_\_stack\_chk\_fail@PLT  .L17:  movq -8(%rbp), %rbx  leave  .cfi\_def\_cfa 7, 8  ret  .cfi\_endproc  .LFE3:  .size main, .-main  .ident "GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",@progbits |

## Bai19\_8086\_TU.s

|  |
| --- |
| .file "b9.c"  .text  .globl MAX  .section .rodata  .align 4  .type MAX, @object  .size MAX, 4  MAX:  .long 100  .LC0:  .string "\nNhap phan tu a[%d] = "  .LC1:  .string "%d"  .text  .globl NhapMang  .type NhapMang, @function  NhapMang:  .LFB0:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 1, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 1  subq $32, %rsp  movq %rdi, -24(%rbp)  movl %esi, -28(%rbp)  movl $0, -4(%rbp)  jmp .L2  .L3:  movl -4(%rbp), %eax  movl %eax, %esi  leaq .LC0(%rip), %rdi  movl $0, %eax  call printf@PLT  movl -4(%rbp), %eax  cltq  leaq 0(,%rax,4), %rdx  movq -24(%rbp), %rax  addq %rdx, %rax  movq %rax, %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  addl $1, -4(%rbp)  .L2:  movl -4(%rbp), %eax  cmpl -28(%rbp), %eax  jl .L3  nop  leave  .cfi\_def\_cfa 2, 8  ret  .cfi\_endproc  .LFE0:  .size NhapMang, .-NhapMang  .section .rodata  .LC2:  .string "\nPhan tu a[%d] = %d"  .text  .globl XuatMang  .type XuatMang, @function  XuatMang:  .LFB1:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 1, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 1  subq $32, %rsp  movq %rdi, -24(%rbp)  movl %esi, -28(%rbp)  movl $0, -4(%rbp)  jmp .L5  .L6:  movl -4(%rbp), %eax  cltq  leaq 0(,%rax,4), %rdx  movq -24(%rbp), %rax  addq %rdx, %rax  movl (%rax), %edx  movl -4(%rbp), %eax  movl %eax, %esi  leaq .LC2(%rip), %rdi  movl $0, %eax  call printf@PLT  addl $1, -4(%rbp)  .L5:  movl -4(%rbp), %eax  cmpl -28(%rbp), %eax  jl .L6  nop  leave  .cfi\_def\_cfa 2, 8  ret  .cfi\_endproc  .LFE1:  .size XuatMang, .-XuatMang  .globl TimKiem  .type TimKiem, @function  TimKiem:  .LFB2:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 1, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 1  movq %rdi, -24(%rbp)  movl %esi, -28(%rbp)  movl %edx, -32(%rbp)  movl $0, -4(%rbp)  jmp .L8  .L11:  movl -4(%rbp), %eax  cltq  leaq 0(,%rax,4), %rdx  movq -24(%rbp), %rax  addq %rdx, %rax  movl (%rax), %eax  cmpl %eax, -32(%rbp)  jne .L9  movl -4(%rbp), %eax  jmp .L10  .L9:  addl $1, -4(%rbp)  .L8:  movl -4(%rbp), %eax  cmpl -28(%rbp), %eax  jl .L11  movl $-1, %eax  .L10:  popq %rbp  .cfi\_def\_cfa 2, 8  ret  .cfi\_endproc  .LFE2:  .size TimKiem, .-TimKiem  .section .rodata  .LC3:  .string "\nNhap so luong phan tu: "  .LC4:  .string "\nNhap lai so luong phan tu: "  .LC5:  .string "\n======NHAP MANG====="  .LC6:  .string "\n======XUAT MANG====="  .text  .globl main  .type main, @function  main:  .LFB3:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 1, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 1  pushq %rbx  subq $40, %rsp  .cfi\_offset 3, -24  movq %fs:40, %rax  movq %rax, -24(%rbp)  xorl %eax, %eax  movq %rsp, %rax  movq %rax, %rbx  movl $100, %eax  cltq  subq $1, %rax  movq %rax, -40(%rbp)  movl $100, %eax  cltq  movq %rax, %rsi  movl $0, %edi  movl $100, %eax  cltq  movq %rax, %rdx  movl $0, %ecx  movl $100, %eax  cltq  salq $2, %rax  leaq 3(%rax), %rdx  movl $16, %eax  subq $1, %rax  addq %rdx, %rax  movl $16, %ecx  movl $0, %edx  divq %rcx  imulq $16, %rax, %rax  subq %rax, %rsp  movq %rsp, %rax  addq $3, %rax  shrq $2, %rax  salq $2, %rax  movq %rax, -32(%rbp)  leaq .LC3(%rip), %rdi  movl $0, %eax  call printf@PLT  .L15:  leaq -44(%rbp), %rax  movq %rax, %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl -44(%rbp), %eax  testl %eax, %eax  jle .L13  movl -44(%rbp), %eax  movl $100, %edx  cmpl %edx, %eax  jle .L14  .L13:  leaq .LC4(%rip), %rdi  movl $0, %eax  call printf@PLT  .L14:  movl -44(%rbp), %eax  testl %eax, %eax  jle .L15  movl -44(%rbp), %eax  movl $100, %edx  cmpl %edx, %eax  jg .L15  leaq .LC5(%rip), %rdi  call puts@PLT  movl -44(%rbp), %edx  movq -32(%rbp), %rax  movl %edx, %esi  movq %rax, %rdi  call NhapMang  leaq .LC6(%rip), %rdi  call puts@PLT  movl -44(%rbp), %edx  movq -32(%rbp), %rax  movl %edx, %esi  movq %rax, %rdi  call XuatMang  movq %rbx, %rsp  movl $0, %eax  movq -24(%rbp), %rcx  xorq %fs:40, %rcx  je .L17  call \_\_stack\_chk\_fail@PLT  .L17:  movq -8(%rbp), %rbx  leave  .cfi\_def\_cfa 2, 8  ret  .cfi\_endproc  .LFE3:  .size main, .-main  .ident "GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",@progbits |

## Bai19\_ARM.s

|  |
| --- |
| .arch armv5t  .eabi\_attribute 20, 1  .eabi\_attribute 21, 1  .eabi\_attribute 23, 3  .eabi\_attribute 24, 1  .eabi\_attribute 25, 1  .eabi\_attribute 26, 2  .eabi\_attribute 30, 6  .eabi\_attribute 34, 0  .eabi\_attribute 18, 4  .file "b9.c"  .text  .global MAX  .section .rodata  .align 2  .type MAX, %object  .size MAX, 4  MAX:  .word 100  .align 2  .LC0:  .ascii "\012Nhap phan tu a[%d] = \000"  .align 2  .LC1:  .ascii "%d\000"  .text  .align 2  .global NhapMang  .syntax unified  .arm  .fpu softvfp  .type NhapMang, %function  NhapMang:  @ args = 0, pretend = 0, frame = 16  @ frame\_needed = 1, uses\_anonymous\_args = 0  push {fp, lr}  add fp, sp, #4  sub sp, sp, #16  str r0, [fp, #-16]  str r1, [fp, #-20]  mov r3, #0  str r3, [fp, #-8]  b .L2  .L3:  ldr r1, [fp, #-8]  ldr r0, .L4  bl printf  ldr r3, [fp, #-8]  lsl r3, r3, #2  ldr r2, [fp, #-16]  add r3, r2, r3  mov r1, r3  ldr r0, .L4+4  bl \_\_isoc99\_scanf  ldr r3, [fp, #-8]  add r3, r3, #1  str r3, [fp, #-8]  .L2:  ldr r2, [fp, #-8]  ldr r3, [fp, #-20]  cmp r2, r3  blt .L3  nop  sub sp, fp, #4  @ sp needed  pop {fp, pc}  .L5:  .align 2  .L4:  .word .LC0  .word .LC1  .size NhapMang, .-NhapMang  .section .rodata  .align 2  .LC2:  .ascii "\012Phan tu a[%d] = %d\000"  .text  .align 2  .global XuatMang  .syntax unified  .arm  .fpu softvfp  .type XuatMang, %function  XuatMang:  @ args = 0, pretend = 0, frame = 16  @ frame\_needed = 1, uses\_anonymous\_args = 0  push {fp, lr}  add fp, sp, #4  sub sp, sp, #16  str r0, [fp, #-16]  str r1, [fp, #-20]  mov r3, #0  str r3, [fp, #-8]  b .L7  .L8:  ldr r3, [fp, #-8]  lsl r3, r3, #2  ldr r2, [fp, #-16]  add r3, r2, r3  ldr r3, [r3]  mov r2, r3  ldr r1, [fp, #-8]  ldr r0, .L9  bl printf  ldr r3, [fp, #-8]  add r3, r3, #1  str r3, [fp, #-8]  .L7:  ldr r2, [fp, #-8]  ldr r3, [fp, #-20]  cmp r2, r3  blt .L8  nop  sub sp, fp, #4  @ sp needed  pop {fp, pc}  .L10:  .align 2  .L9:  .word .LC2  .size XuatMang, .-XuatMang  .align 2  .global TimKiem  .syntax unified  .arm  .fpu softvfp  .type TimKiem, %function  TimKiem:  @ args = 0, pretend = 0, frame = 24  @ frame\_needed = 1, uses\_anonymous\_args = 0  @ link register save eliminated.  str fp, [sp, #-4]!  add fp, sp, #0  sub sp, sp, #28  str r0, [fp, #-16]  str r1, [fp, #-20]  str r2, [fp, #-24]  mov r3, #0  str r3, [fp, #-8]  b .L12  .L15:  ldr r3, [fp, #-8]  lsl r3, r3, #2  ldr r2, [fp, #-16]  add r3, r2, r3  ldr r3, [r3]  ldr r2, [fp, #-24]  cmp r2, r3  bne .L13  ldr r3, [fp, #-8]  b .L14  .L13:  ldr r3, [fp, #-8]  add r3, r3, #1  str r3, [fp, #-8]  .L12:  ldr r2, [fp, #-8]  ldr r3, [fp, #-20]  cmp r2, r3  blt .L15  mvn r3, #0  .L14:  mov r0, r3  add sp, fp, #0  @ sp needed  ldr fp, [sp], #4  bx lr  .size TimKiem, .-TimKiem  .section .rodata  .align 2  .LC3:  .ascii "\012Nhap so luong phan tu: \000"  .align 2  .LC4:  .ascii "\012Nhap lai so luong phan tu: \000"  .align 2  .LC5:  .ascii "\012======NHAP MANG=====\000"  .align 2  .LC6:  .ascii "\012======XUAT MANG=====\000"  .text  .align 2  .global main  .syntax unified  .arm  .fpu softvfp  .type main, %function  main:  @ args = 0, pretend = 0, frame = 16  @ frame\_needed = 1, uses\_anonymous\_args = 0  push {r4, r5, fp, lr}  add fp, sp, #12  sub sp, sp, #16  ldr r3, .L22  ldr r3, [r3]  str r3, [fp, #-16]  mov r3, sp  mov r5, r3  mov r3, #100  sub r3, r3, #1  str r3, [fp, #-24]  mov r3, #100  mov r1, r3  mov r2, #0  mov r3, #0  mov r4, #0  lsl r4, r2, #5  orr r4, r4, r1, lsr #27  lsl r3, r1, #5  mov r3, #100  mov r1, r3  mov r2, #0  mov r3, #0  mov r4, #0  lsl r4, r2, #5  orr r4, r4, r1, lsr #27  lsl r3, r1, #5  mov r3, #100  lsl r3, r3, #2  add r3, r3, #3  add r3, r3, #7  lsr r3, r3, #3  lsl r3, r3, #3  sub sp, sp, r3  mov r3, sp  add r3, r3, #3  lsr r3, r3, #2  lsl r3, r3, #2  str r3, [fp, #-20]  ldr r0, .L22+4  bl printf  .L19:  sub r3, fp, #28  mov r1, r3  ldr r0, .L22+8  bl \_\_isoc99\_scanf  ldr r3, [fp, #-28]  cmp r3, #0  ble .L17  ldr r3, [fp, #-28]  mov r2, #100  cmp r3, r2  ble .L18  .L17:  ldr r0, .L22+12  bl printf  .L18:  ldr r3, [fp, #-28]  cmp r3, #0  ble .L19  ldr r3, [fp, #-28]  mov r2, #100  cmp r3, r2  bgt .L19  ldr r0, .L22+16  bl puts  ldr r3, [fp, #-20]  ldr r2, [fp, #-28]  mov r1, r2  mov r0, r3  bl NhapMang  ldr r0, .L22+20  bl puts  ldr r3, [fp, #-20]  ldr r2, [fp, #-28]  mov r1, r2  mov r0, r3  bl XuatMang  mov sp, r5  mov r3, #0  mov r0, r3  ldr r3, .L22  ldr r2, [fp, #-16]  ldr r3, [r3]  cmp r2, r3  beq .L21  bl \_\_stack\_chk\_fail  .L21:  sub sp, fp, #12  @ sp needed  pop {r4, r5, fp, pc}  .L23:  .align 2  .L22:  .word \_\_stack\_chk\_guard  .word .LC3  .word .LC1  .word .LC4  .word .LC5  .word .LC6  .size main, .-main  .ident "GCC: (Ubuntu/Linaro 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",%progbits |

## Bai19\_ARM\_TU.s

|  |
| --- |
| .arch armv5t  .eabi\_attribute 20, 1  .eabi\_attribute 21, 1  .eabi\_attribute 23, 3  .eabi\_attribute 24, 1  .eabi\_attribute 25, 1  .eabi\_attribute 26, 2  .eabi\_attribute 30, 2  .eabi\_attribute 34, 0  .eabi\_attribute 18, 4  .file "b9.c"  .text  .section .rodata.str1.4,"aMS",%progbits,1  .align 2  .LC0:  .ascii "\012Nhap phan tu a[%d] = \000"  .align 2  .LC1:  .ascii "%d\000"  .text  .align 2  .global NhapMang  .syntax unified  .arm  .fpu softvfp  .type NhapMang, %function  NhapMang:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 0, uses\_anonymous\_args = 0  push {r4, r5, r6, r7, r8, lr}  subs r6, r1, #0  pople {r4, r5, r6, r7, r8, pc}  mov r5, r0  mov r4, #0  ldr r8, .L7  ldr r7, .L7+4  .L3:  mov r0, #1  mov r2, r4  mov r1, r8  add r4, r4, r0  bl \_\_printf\_chk  mov r1, r5  mov r0, r7  bl \_\_isoc99\_scanf  cmp r6, r4  add r5, r5, #4  bne .L3  pop {r4, r5, r6, r7, r8, pc}  .L8:  .align 2  .L7:  .word .LC0  .word .LC1  .size NhapMang, .-NhapMang  .section .rodata.str1.4  .align 2  .LC2:  .ascii "\012Phan tu a[%d] = %d\000"  .text  .align 2  .global XuatMang  .syntax unified  .arm  .fpu softvfp  .type XuatMang, %function  XuatMang:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 0, uses\_anonymous\_args = 0  push {r4, r5, r6, r7, r8, lr}  subs r6, r1, #0  pople {r4, r5, r6, r7, r8, pc}  mov r4, #0  ldr r7, .L14  sub r5, r0, #4  .L11:  mov r2, r4  mov r1, r7  mov r0, #1  ldr r3, [r5, #4]!  add r4, r4, #1  bl \_\_printf\_chk  cmp r6, r4  bne .L11  pop {r4, r5, r6, r7, r8, pc}  .L15:  .align 2  .L14:  .word .LC2  .size XuatMang, .-XuatMang  .section .rodata.str1.4  .align 2  .LC3:  .ascii "\012Nhap so luong phan tu: \000"  .align 2  .LC4:  .ascii "\012Nhap lai so luong phan tu: \000"  .align 2  .LC5:  .ascii "\012======NHAP MANG=====\000"  .align 2  .LC6:  .ascii "\012======XUAT MANG=====\000"  .section .rodata.cst4,"aM",%progbits,4  .align 2  .LC7:  .word \_\_stack\_chk\_guard  .section .text.startup,"ax",%progbits  .align 2  .global main  .syntax unified  .arm  .fpu softvfp  .type main, %function  main:  @ args = 0, pretend = 0, frame = 8  @ frame\_needed = 1, uses\_anonymous\_args = 0  push {r4, r5, r6, fp, lr}  add fp, sp, #16  sub sp, sp, #12  ldr r3, .L24  sub sp, sp, #400  mov r0, #1  ldr r1, .L24+4  ldr r3, [r3]  str r3, [fp, #-24]  mov r3,#0  mov r4, sp  bl \_\_printf\_chk  ldr r5, .L24+8  ldr r6, .L24+12  .L18:  mov r0, r5  sub r1, fp, #28  bl \_\_isoc99\_scanf  ldr r3, [fp, #-28]  sub r3, r3, #1  cmp r3, #99  bhi .L22  .L17:  ldr r0, .L24+16  bl puts  ldr r1, [fp, #-28]  mov r0, r4  bl NhapMang  ldr r0, .L24+20  bl puts  mov r0, r4  ldr r1, [fp, #-28]  bl XuatMang  ldr r3, .L24  ldr r2, [r3]  ldr r3, [fp, #-24]  eors r2, r3, r2  mov r3, #0  bne .L23  mov r0, #0  sub sp, fp, #16  @ sp needed  pop {r4, r5, r6, fp, pc}  .L22:  mov r1, r6  mov r0, #1  bl \_\_printf\_chk  ldr r3, [fp, #-28]  sub r3, r3, #1  cmp r3, #99  bhi .L18  b .L17  .L23:  bl \_\_stack\_chk\_fail  .L25:  .align 2  .L24:  .word .LC7  .word .LC3  .word .LC1  .word .LC4  .word .LC5  .word .LC6  .size main, .-main  .global MAX  .section .rodata  .align 2  .type MAX, %object  .size MAX, 4  MAX:  .word 100  .ident "GCC: (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0"  .section .note.GNU-stack,"",%progbits |

# Vd20:

## Bai20.c

|  |
| --- |
| #include <stdio.h>    const int MAX = 100;      void NhapMang(int a[], int n){  for(int i = 0;i < n; ++i){  printf("\nNhap phan tu a[%d] = ", i);  scanf("%d", &a[i]);  }  }    int TimKiem(int a[], int n, int v){  for(int i = 0;i < n; ++i){  if(a[i] == v){  return i;  }  }  return -1;  }    int main(){  int arr[MAX];  int n;  printf("\nNhap so luong phan tu: ");  do{  scanf("%d", &n);  if(n <= 0 || n > MAX){  printf("\nNhap lai so luong phan tu: ");  }  }while(n <= 0 || n > MAX);  printf("\n======NHAP MANG=====\n");  NhapMang(arr, n);    int v;  printf("\nNhap vao gia tri can tim: ");  scanf("%d", &v);  printf("\nTim thay so %d tai chi so %d!", v, TimKiem(arr, n, v));  } |

## Bai20\_TU.c

|  |
| --- |
| #include<stdio.h>  void main()  {  int a[50];  int i,n;  printf("Nhap so phan tu mang: ");  scanf("%d",&n);  for(i=0;i<n;i++)  {  scanf("%d",&a[i]);  }  int v;  printf("\nNhap vao gia tri can tim: ");  scanf("%d", &v);  int j=0;  for( j =0;j < n; ++j){  if(a[j] == v){  printf("\nTim thay so %d tai chi so %d!", v, j);  }  }      } |

## Bai20\_8086.s

|  |
| --- |
| .file "b10.c"  .text  .globl MAX  .section .rodata  .align 4  .type MAX, @object  .size MAX, 4  MAX:  .long 100  .LC0:  .string "\nNhap phan tu a[%d] = "  .LC1:  .string "%d"  .text  .globl NhapMang  .type NhapMang, @function  NhapMang:  .LFB0:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 6, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 6  subq $32, %rsp  movq %rdi, -24(%rbp)  movl %esi, -28(%rbp)  movl $0, -4(%rbp)  jmp .L2  .L3:  movl -4(%rbp), %eax  movl %eax, %esi  leaq .LC0(%rip), %rdi  movl $0, %eax  call printf@PLT  movl -4(%rbp), %eax  cltq  leaq 0(,%rax,4), %rdx  movq -24(%rbp), %rax  addq %rdx, %rax  movq %rax, %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  addl $1, -4(%rbp)  .L2:  movl -4(%rbp), %eax  cmpl -28(%rbp), %eax  jl .L3  nop  leave  .cfi\_def\_cfa 7, 8  ret  .cfi\_endproc  .LFE0:  .size NhapMang, .-NhapMang  .globl TimKiem  .type TimKiem, @function  TimKiem:  .LFB1:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 6, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 6  movq %rdi, -24(%rbp)  movl %esi, -28(%rbp)  movl %edx, -32(%rbp)  movl $0, -4(%rbp)  jmp .L5  .L8:  movl -4(%rbp), %eax  cltq  leaq 0(,%rax,4), %rdx  movq -24(%rbp), %rax  addq %rdx, %rax  movl (%rax), %eax  cmpl %eax, -32(%rbp)  jne .L6  movl -4(%rbp), %eax  jmp .L7  .L6:  addl $1, -4(%rbp)  .L5:  movl -4(%rbp), %eax  cmpl -28(%rbp), %eax  jl .L8  movl $-1, %eax  .L7:  popq %rbp  .cfi\_def\_cfa 7, 8  ret  .cfi\_endproc  .LFE1:  .size TimKiem, .-TimKiem  .section .rodata  .LC2:  .string "\nNhap so luong phan tu: "  .LC3:  .string "\nNhap lai so luong phan tu: "  .LC4:  .string "\n======NHAP MANG====="  .LC5:  .string "\nNhap vao gia tri can tim: "  .align 8  .LC6:  .string "\nTim thay so %d tai chi so %d!"  .text  .globl main  .type main, @function  main:  .LFB2:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 6, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 6  pushq %rbx  subq $40, %rsp  .cfi\_offset 3, -24  movq %fs:40, %rax  movq %rax, -24(%rbp)  xorl %eax, %eax  movq %rsp, %rax  movq %rax, %rbx  movl $100, %eax  cltq  subq $1, %rax  movq %rax, -40(%rbp)  movl $100, %eax  cltq  movq %rax, %rsi  movl $0, %edi  movl $100, %eax  cltq  movq %rax, %rdx  movl $0, %ecx  movl $100, %eax  cltq  salq $2, %rax  leaq 3(%rax), %rdx  movl $16, %eax  subq $1, %rax  addq %rdx, %rax  movl $16, %ecx  movl $0, %edx  divq %rcx  imulq $16, %rax, %rax  subq %rax, %rsp  movq %rsp, %rax  addq $3, %rax  shrq $2, %rax  salq $2, %rax  movq %rax, -32(%rbp)  leaq .LC2(%rip), %rdi  movl $0, %eax  call printf@PLT  .L12:  leaq -48(%rbp), %rax  movq %rax, %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl -48(%rbp), %eax  testl %eax, %eax  jle .L10  movl -48(%rbp), %eax  movl $100, %edx  cmpl %edx, %eax  jle .L11  .L10:  leaq .LC3(%rip), %rdi  movl $0, %eax  call printf@PLT  .L11:  movl -48(%rbp), %eax  testl %eax, %eax  jle .L12  movl -48(%rbp), %eax  movl $100, %edx  cmpl %edx, %eax  jg .L12  leaq .LC4(%rip), %rdi  call puts@PLT  movl -48(%rbp), %edx  movq -32(%rbp), %rax  movl %edx, %esi  movq %rax, %rdi  call NhapMang  leaq .LC5(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq -44(%rbp), %rax  movq %rax, %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl -44(%rbp), %edx  movl -48(%rbp), %ecx  movq -32(%rbp), %rax  movl %ecx, %esi  movq %rax, %rdi  call TimKiem  movl %eax, %edx  movl -44(%rbp), %eax  movl %eax, %esi  leaq .LC6(%rip), %rdi  movl $0, %eax  call printf@PLT  movq %rbx, %rsp  movl $0, %eax  movq -24(%rbp), %rcx  xorq %fs:40, %rcx  je .L14  call \_\_stack\_chk\_fail@PLT  .L14:  movq -8(%rbp), %rbx  leave  .cfi\_def\_cfa 7, 8  ret  .cfi\_endproc  .LFE2:  .size main, .-main  .ident "GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",@progbits |

## Bai20\_8086\_TU.s

|  |
| --- |
| .file "b10.c"  .text  .globl MAX  .section .rodata  .align 4  .type MAX, @object  .size MAX, 4  MAX:  .long 100  .LC0:  .string "\nNhap phan tu a[%d] = "  .LC1:  .string "%d"  .text  .globl NhapMang  .type NhapMang, @function  NhapMang:  .LFB0:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 1, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 1  subq $32, %rsp  movq %rdi, -24(%rbp)  movl %esi, -28(%rbp)  movl $0, -4(%rbp)  jmp .L2  .L3:  movl -4(%rbp), %eax  movl %eax, %esi  leaq .LC0(%rip), %rdi  movl $0, %eax  call printf@PLT  movl -4(%rbp), %eax  cltq  leaq 0(,%rax,4), %rdx  movq -24(%rbp), %rax  addq %rdx, %rax  movq %rax, %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  addl $1, -4(%rbp)  .L2:  movl -4(%rbp), %eax  cmpl -28(%rbp), %eax  jl .L3  nop  leave  .cfi\_def\_cfa 2, 8  ret  .cfi\_endproc  .LFE0:  .size NhapMang, .-NhapMang  .globl TimKiem  .type TimKiem, @function  TimKiem:  .LFB1:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 1, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 1  movq %rdi, -24(%rbp)  movl %esi, -28(%rbp)  movl %edx, -32(%rbp)  movl $0, -4(%rbp)  jmp .L5  .L8:  movl -4(%rbp), %eax  cltq  leaq 0(,%rax,4), %rdx  movq -24(%rbp), %rax  addq %rdx, %rax  movl (%rax), %eax  cmpl %eax, -32(%rbp)  jne .L6  movl -4(%rbp), %eax  jmp .L7  .L6:  addl $1, -4(%rbp)  .L5:  movl -4(%rbp), %eax  cmpl -28(%rbp), %eax  jl .L8  movl $-1, %eax  .L7:  popq %rbp  .cfi\_def\_cfa 2, 8  ret  .cfi\_endproc  .LFE1:  .size TimKiem, .-TimKiem  .section .rodata  .LC2:  .string "\nNhap so luong phan tu: "  .LC3:  .string "\nNhap lai so luong phan tu: "  .LC4:  .string "\n======NHAP MANG====="  .LC5:  .string "\nNhap vao gia tri can tim: "  .align 8  .LC6:  .string "\nTim thay so %d tai chi so %d!"  .text  .globl main  .type main, @function  main:  .LFB2:  .cfi\_startproc  pushq %rbp  .cfi\_def\_cfa\_offset 16  .cfi\_offset 1, -16  movq %rsp, %rbp  .cfi\_def\_cfa\_register 1  pushq %rbx  subq $40, %rsp  .cfi\_offset 3, -24  movq %fs:40, %rax  movq %rax, -24(%rbp)  xorl %eax, %eax  movq %rsp, %rax  movq %rax, %rbx  movl $100, %eax  cltq  subq $1, %rax  movq %rax, -40(%rbp)  movl $100, %eax  cltq  movq %rax, %rsi  movl $0, %edi  movl $100, %eax  cltq  movq %rax, %rdx  movl $0, %ecx  movl $100, %eax  cltq  salq $2, %rax  leaq 3(%rax), %rdx  movl $16, %eax  subq $1, %rax  addq %rdx, %rax  movl $16, %ecx  movl $0, %edx  divq %rcx  imulq $16, %rax, %rax  subq %rax, %rsp  movq %rsp, %rax  addq $3, %rax  shrq $2, %rax  salq $2, %rax  movq %rax, -32(%rbp)  leaq .LC2(%rip), %rdi  movl $0, %eax  call printf@PLT  .L12:  leaq -48(%rbp), %rax  movq %rax, %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl -48(%rbp), %eax  testl %eax, %eax  jle .L10  movl -48(%rbp), %eax  movl $100, %edx  cmpl %edx, %eax  jle .L11  .L10:  leaq .LC3(%rip), %rdi  movl $0, %eax  call printf@PLT  .L11:  movl -48(%rbp), %eax  testl %eax, %eax  jle .L12  movl -48(%rbp), %eax  movl $100, %edx  cmpl %edx, %eax  jg .L12  leaq .LC4(%rip), %rdi  call puts@PLT  movl -48(%rbp), %edx  movq -32(%rbp), %rax  movl %edx, %esi  movq %rax, %rdi  call NhapMang  leaq .LC5(%rip), %rdi  movl $0, %eax  call printf@PLT  leaq -44(%rbp), %rax  movq %rax, %rsi  leaq .LC1(%rip), %rdi  movl $0, %eax  call \_\_isoc99\_scanf@PLT  movl -44(%rbp), %edx  movl -48(%rbp), %ecx  movq -32(%rbp), %rax  movl %ecx, %esi  movq %rax, %rdi  call TimKiem  movl %eax, %edx  movl -44(%rbp), %eax  movl %eax, %esi  leaq .LC6(%rip), %rdi  movl $0, %eax  call printf@PLT  movq %rbx, %rsp  movl $0, %eax  movq -24(%rbp), %rcx  xorq %fs:40, %rcx  je .L14  call \_\_stack\_chk\_fail@PLT  .L14:  movq -8(%rbp), %rbx  leave  .cfi\_def\_cfa 2, 8  ret  .cfi\_endproc  .LFE2:  .size main, .-main  .ident "GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",@progbits |

## Bai20\_ARM.s

|  |
| --- |
| .arch armv5t  .eabi\_attribute 20, 1  .eabi\_attribute 21, 1  .eabi\_attribute 23, 3  .eabi\_attribute 24, 1  .eabi\_attribute 25, 1  .eabi\_attribute 26, 2  .eabi\_attribute 30, 6  .eabi\_attribute 34, 0  .eabi\_attribute 18, 4  .file "b10.c"  .text  .global MAX  .section .rodata  .align 2  .type MAX, %object  .size MAX, 4  MAX:  .word 100  .align 2  .LC0:  .ascii "\012Nhap phan tu a[%d] = \000"  .align 2  .LC1:  .ascii "%d\000"  .text  .align 2  .global NhapMang  .syntax unified  .arm  .fpu softvfp  .type NhapMang, %function  NhapMang:  @ args = 0, pretend = 0, frame = 16  @ frame\_needed = 1, uses\_anonymous\_args = 0  push {fp, lr}  add fp, sp, #4  sub sp, sp, #16  str r0, [fp, #-16]  str r1, [fp, #-20]  mov r3, #0  str r3, [fp, #-8]  b .L2  .L3:  ldr r1, [fp, #-8]  ldr r0, .L4  bl printf  ldr r3, [fp, #-8]  lsl r3, r3, #2  ldr r2, [fp, #-16]  add r3, r2, r3  mov r1, r3  ldr r0, .L4+4  bl \_\_isoc99\_scanf  ldr r3, [fp, #-8]  add r3, r3, #1  str r3, [fp, #-8]  .L2:  ldr r2, [fp, #-8]  ldr r3, [fp, #-20]  cmp r2, r3  blt .L3  nop  sub sp, fp, #4  @ sp needed  pop {fp, pc}  .L5:  .align 2  .L4:  .word .LC0  .word .LC1  .size NhapMang, .-NhapMang  .align 2  .global TimKiem  .syntax unified  .arm  .fpu softvfp  .type TimKiem, %function  TimKiem:  @ args = 0, pretend = 0, frame = 24  @ frame\_needed = 1, uses\_anonymous\_args = 0  @ link register save eliminated.  str fp, [sp, #-4]!  add fp, sp, #0  sub sp, sp, #28  str r0, [fp, #-16]  str r1, [fp, #-20]  str r2, [fp, #-24]  mov r3, #0  str r3, [fp, #-8]  b .L7  .L10:  ldr r3, [fp, #-8]  lsl r3, r3, #2  ldr r2, [fp, #-16]  add r3, r2, r3  ldr r3, [r3]  ldr r2, [fp, #-24]  cmp r2, r3  bne .L8  ldr r3, [fp, #-8]  b .L9  .L8:  ldr r3, [fp, #-8]  add r3, r3, #1  str r3, [fp, #-8]  .L7:  ldr r2, [fp, #-8]  ldr r3, [fp, #-20]  cmp r2, r3  blt .L10  mvn r3, #0  .L9:  mov r0, r3  add sp, fp, #0  @ sp needed  ldr fp, [sp], #4  bx lr  .size TimKiem, .-TimKiem  .section .rodata  .align 2  .LC2:  .ascii "\012Nhap so luong phan tu: \000"  .align 2  .LC3:  .ascii "\012Nhap lai so luong phan tu: \000"  .align 2  .LC4:  .ascii "\012======NHAP MANG=====\000"  .align 2  .LC5:  .ascii "\012Nhap vao gia tri can tim: \000"  .align 2  .LC6:  .ascii "\012Tim thay so %d tai chi so %d!\000"  .text  .align 2  .global main  .syntax unified  .arm  .fpu softvfp  .type main, %function  main:  @ args = 0, pretend = 0, frame = 24  @ frame\_needed = 1, uses\_anonymous\_args = 0  push {r4, r5, fp, lr}  add fp, sp, #12  sub sp, sp, #24  ldr r3, .L17  ldr r3, [r3]  str r3, [fp, #-16]  mov r3, sp  mov r5, r3  mov r3, #100  sub r3, r3, #1  str r3, [fp, #-24]  mov r3, #100  mov r1, r3  mov r2, #0  mov r3, #0  mov r4, #0  lsl r4, r2, #5  orr r4, r4, r1, lsr #27  lsl r3, r1, #5  mov r3, #100  mov r1, r3  mov r2, #0  mov r3, #0  mov r4, #0  lsl r4, r2, #5  orr r4, r4, r1, lsr #27  lsl r3, r1, #5  mov r3, #100  lsl r3, r3, #2  add r3, r3, #3  add r3, r3, #7  lsr r3, r3, #3  lsl r3, r3, #3  sub sp, sp, r3  mov r3, sp  add r3, r3, #3  lsr r3, r3, #2  lsl r3, r3, #2  str r3, [fp, #-20]  ldr r0, .L17+4  bl printf  .L14:  sub r3, fp, #32  mov r1, r3  ldr r0, .L17+8  bl \_\_isoc99\_scanf  ldr r3, [fp, #-32]  cmp r3, #0  ble .L12  ldr r3, [fp, #-32]  mov r2, #100  cmp r3, r2  ble .L13  .L12:  ldr r0, .L17+12  bl printf  .L13:  ldr r3, [fp, #-32]  cmp r3, #0  ble .L14  ldr r3, [fp, #-32]  mov r2, #100  cmp r3, r2  bgt .L14  ldr r0, .L17+16  bl puts  ldr r3, [fp, #-20]  ldr r2, [fp, #-32]  mov r1, r2  mov r0, r3  bl NhapMang  ldr r0, .L17+20  bl printf  sub r3, fp, #28  mov r1, r3  ldr r0, .L17+8  bl \_\_isoc99\_scanf  ldr r4, [fp, #-28]  ldr r3, [fp, #-20]  ldr r1, [fp, #-32]  ldr r2, [fp, #-28]  mov r0, r3  bl TimKiem  mov r3, r0  mov r2, r3  mov r1, r4  ldr r0, .L17+24  bl printf  mov sp, r5  mov r3, #0  mov r0, r3  ldr r3, .L17  ldr r2, [fp, #-16]  ldr r3, [r3]  cmp r2, r3  beq .L16  bl \_\_stack\_chk\_fail  .L16:  sub sp, fp, #12  @ sp needed  pop {r4, r5, fp, pc}  .L18:  .align 2  .L17:  .word \_\_stack\_chk\_guard  .word .LC2  .word .LC1  .word .LC3  .word .LC4  .word .LC5  .word .LC6  .size main, .-main  .ident "GCC: (Ubuntu/Linaro 7.5.0-3ubuntu1~18.04) 7.5.0"  .section .note.GNU-stack,"",%progbits |

## Bai20\_ARM\_TU.s

|  |
| --- |
| .arch armv5t  .eabi\_attribute 20, 1  .eabi\_attribute 21, 1  .eabi\_attribute 23, 3  .eabi\_attribute 24, 1  .eabi\_attribute 25, 1  .eabi\_attribute 26, 2  .eabi\_attribute 30, 2  .eabi\_attribute 34, 0  .eabi\_attribute 18, 4  .file "b10.c"  .text  .section .rodata.str1.4,"aMS",%progbits,1  .align 2  .LC0:  .ascii "\012Nhap phan tu a[%d] = \000"  .align 2  .LC1:  .ascii "%d\000"  .text  .align 2  .global NhapMang  .syntax unified  .arm  .fpu softvfp  .type NhapMang, %function  NhapMang:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 0, uses\_anonymous\_args = 0  push {r4, r5, r6, r7, r8, lr}  subs r6, r1, #0  pople {r4, r5, r6, r7, r8, pc}  mov r5, r0  mov r4, #0  ldr r8, .L7  ldr r7, .L7+4  .L3:  mov r0, #1  mov r2, r4  mov r1, r8  add r4, r4, r0  bl \_\_printf\_chk  mov r1, r5  mov r0, r7  bl \_\_isoc99\_scanf  cmp r6, r4  add r5, r5, #4  bne .L3  pop {r4, r5, r6, r7, r8, pc}  .L8:  .align 2  .L7:  .word .LC0  .word .LC1  .size NhapMang, .-NhapMang  .align 2  .global TimKiem  .syntax unified  .arm  .fpu softvfp  .type TimKiem, %function  TimKiem:  @ args = 0, pretend = 0, frame = 0  @ frame\_needed = 0, uses\_anonymous\_args = 0  @ link register save eliminated.  cmp r1, #0  mov r3, r0  ble .L12  mov r0, #0  sub r3, r3, #4  .L11:  ldr ip, [r3, #4]!  cmp ip, r2  bxeq lr  add r0, r0, #1  cmp r1, r0  bne .L11  .L12:  mvn r0, #0  bx lr  .size TimKiem, .-TimKiem  .section .rodata.str1.4  .align 2  .LC2:  .ascii "\012Nhap so luong phan tu: \000"  .align 2  .LC3:  .ascii "\012Nhap lai so luong phan tu: \000"  .align 2  .LC4:  .ascii "\012======NHAP MANG=====\000"  .align 2  .LC5:  .ascii "\012Nhap vao gia tri can tim: \000"  .align 2  .LC6:  .ascii "\012Tim thay so %d tai chi so %d!\000"  .section .rodata.cst4,"aM",%progbits,4  .align 2  .LC7:  .word \_\_stack\_chk\_guard  .section .text.startup,"ax",%progbits  .align 2  .global main  .syntax unified  .arm  .fpu softvfp  .type main, %function  main:  @ args = 0, pretend = 0, frame = 16  @ frame\_needed = 1, uses\_anonymous\_args = 0  push {r4, r5, r6, fp, lr}  add fp, sp, #16  sub sp, sp, #20  ldr r3, .L27  sub sp, sp, #400  mov r0, #1  ldr r1, .L27+4  ldr r3, [r3]  str r3, [fp, #-24]  mov r3,#0  mov r4, sp  bl \_\_printf\_chk  ldr r5, .L27+8  ldr r6, .L27+12  .L16:  mov r0, r5  sub r1, fp, #32  bl \_\_isoc99\_scanf  ldr r3, [fp, #-32]  sub r3, r3, #1  cmp r3, #99  bhi .L24  .L15:  ldr r0, .L27+16  bl puts  mov r0, r4  ldr r1, [fp, #-32]  bl NhapMang  ldr r1, .L27+20  mov r0, #1  bl \_\_printf\_chk  sub r1, fp, #28  ldr r0, .L27+8  bl \_\_isoc99\_scanf  ldr r1, [fp, #-32]  ldr r2, [fp, #-28]  cmp r1, #0  ble .L20  mov r3, #0  sub r0, r4, #4  b .L18  .L25:  add r3, r3, #1  cmp r1, r3  beq .L20  .L18:  ldr ip, [r0, #4]!  cmp r2, ip  bne .L25  .L17:  mov r0, #1  ldr r1, .L27+24  bl \_\_printf\_chk  ldr r3, .L27  ldr r2, [r3]  ldr r3, [fp, #-24]  eors r2, r3, r2  mov r3, #0  bne .L26  mov r0, #0  sub sp, fp, #16  @ sp needed  pop {r4, r5, r6, fp, pc}  .L24:  mov r1, r6  mov r0, #1  bl \_\_printf\_chk  ldr r3, [fp, #-32]  sub r3, r3, #1  cmp r3, #99  bhi .L16  b .L15  .L20:  mvn r3, #0  b .L17  .L26:  bl \_\_stack\_chk\_fail  .L28:  .align 2  .L27:  .word .LC7  .word .LC2  .word .LC1  .word .LC3  .word .LC4  .word .LC5  .word .LC6  .size main, .-main  .global MAX  .section .rodata  .align 2  .type MAX, %object  .size MAX, 4  MAX:  .word 100  .ident "GCC: (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0"  .section .note.GNU-stack,"",%progbits |

**BẢNG 1. Tối ưu tốc độ giai đoạn lập trình kết hợp tối ưu trình biên dịch GCC**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên chương trình** | **Kỹ thuật tối ưu thực hiện** | **Ct gốc** | **Ct tối ưu** |
| 11 | Tính tổng bình phương các số lẻ từ 1 đến < n | Rút gọn câu lệnh, thay đổi câu khai báo. | 2.204s | 1.1475s |
| 12 | Tìm max của 3 số a,b,c nhập từ bàn phím | Rút gọn câu lệnh, thay đổi câu khai báo và loại bỏ hàm con. | 2.104 | 1.576 |
| 13 | Nhập một số nguyên từ bàn phím, kiểm tra xem đó là số chẵn hay lẻ | Rút gọn câu lệnh, thay đổi câu khai báo. | 1.021 | 0.592 |
| 14 | Tìm ước số chung lớn nhất và bội số chung nhỏ nhất của 2 số nguyên nhập từ bàn phím | Rút gọn câu lệnh, thay đổi câu khai báo. Ko dùng biến global | 3.499 | 3.165 |
| 15 | Nhập một số nguyên từ bàn phím. Kiểm tra một số có phải là số hoàn hảo? | Rút gọn chương trình, ko dùng hàm con | 1.628 | 1.217 |
| 16 | Nhập vào dãy số tính tổng các số lẻ trong dãy. | Rút gọn lệnh , không dùng hàm con | 3.049 | 2.221 |
| 17 | Nhập vào 1 mảng và in các số nguyên tố trong mảng và vị trí của nó | Rút gọn lệnh , không dùng biến global | 2.013 | 1.462 |
| 18 | Nhập vào 1 dãy số và in ra vị trí lớn nhất | Rút gọn lệnh , không dùng biến global | 2.144 | 1.823 |
| 19 | Nhập vào 1 mảng và xuất mảng đó ra console. | Rút gọn lệnh , không dùng hàm con | 2.401 | 1.666 |
| 20 | Nhập vào 1 mảng và tìm kiếm 1 giá trị bất kỳ trong mảng và in ra. | Rút gọn lệnh , không dùng hàm con | 4.208 | 2.988 |

**BẢNG 2. Tối ưu kích thước chương trình**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Tên chương trình** | **Mô tả chương trình** | **Kỹ thuật tối ưu thực hiện** | **Ct gốc** | **Ct tối ưu** |
| 11 | Tính tổng bình phương các số lẻ từ 1 đến < n |  | Rút gọn câu lệnh, thay đổi câu khai báo. | 8424 | 8400 |
| 12 | Tìm max của 3 số a,b,c nhập từ bàn phím |  | Rút gọn câu lệnh, thay đổi câu khai báo và loại bỏ hàm con. | 8480 | 8400 |
| 13 | Nhập một số nguyên từ bàn phím, kiểm tra xem đó là số chẵn hay lẻ |  | Rút gọn câu lệnh, thay đổi câu khai báo. | 8376 | 8284 |
| 14 | Tìm ước số chung lớn nhất và bội số chung nhỏ nhất của 2 số nguyên nhập từ bàn phím |  | Rút gọn câu lệnh, thay đổi câu khai báo. Ko dùng biến global | 8448 | 8386 |
| 15 | Nhập một số nguyên từ bàn phím. Kiểm tra một số có phải là số hoàn hảo? |  | Rút gọn chương trình, ko dùng hàm con | 8496 | 8448 |
| 16 | Nhập vào dãy số tính tổng các số lẻ trong dãy. |  | Rút gọn lệnh , không dùng hàm con | 8504 | 8400 |
| 17 | Nhập vào 1 mảng và in các số nguyên tố trong mảng và vị trí của nó |  | Rút gọn lệnh , không dùng biến global | 8472 | 8400 |
| 18 | Nhập vào 1 dãy số và in ra vị trí lớn nhất |  | Rút gọn lệnh , không dùng biến global | 8528 | 8400 |
| 19 | Nhập vào 1 mảng và xuất mảng đó ra console. |  | Rút gọn lệnh , không dùng hàm con | 8568 | 8400 |
| 20 | Nhập vào 1 mảng và tìm kiếm 1 giá trị bất kỳ trong mảng và in ra. |  | Rút gọn lệnh , không dùng hàm con | 8536 | 8408 |

**BẢNG 3. Tối ưu thời gian chương trình** **arm**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Tên chương trình** | **Mô tả chương trình** | **Kỹ thuật tối ưu thực hiện** | **Ct gốc** | **Ct tối ưu** |
| 11 | Tính tổng bình phương các số lẻ từ 1 đến < n |  | Dùng cờ O2 | 0.016s | 0.031s |
| 12 | Tìm max của 3 số a,b,c nhập từ bàn phím |  | Dùng cờ O2 | 0.016s | 0.016s |
| 13 | Nhập một số nguyên từ bàn phím, kiểm tra xem đó là số chẵn hay lẻ |  | Dùng cờ O2 | 0.016s | 0.016s |
| 14 | Tìm ước số chung lớn nhất và bội số chung nhỏ nhất của 2 số nguyên nhập từ bàn phím |  | Dùng cờ O2 | 0.016s | 0.000s |
| 15 | Nhập một số nguyên từ bàn phím. Kiểm tra một số có phải là số hoàn hảo? |  | Dùng cờ O2 | 0.031s | 0.016s |
| 16 | Nhập vào dãy số tính tổng các số lẻ trong dãy. |  | Dùng cờ O2 | 0.031s | 0.016s |
| 17 | Nhập vào 1 mảng và in các số nguyên tố trong mảng và vị trí của nó |  | Dùng cờ O2 | 0.016s | 0.031s |
| 18 | Nhập vào 1 dãy số và in ra vị trí lớn nhất |  | Dùng cờ O2 | 0.031s | 0.016s |
| 19 | Nhập vào 1 mảng và xuất mảng đó ra console. |  | Dùng cờ O2 | 0.031s | 0.016s |
| 20 | Nhập vào 1 mảng và tìm kiếm 1 giá trị bất kỳ trong mảng và in ra. |  | Dùng cờ O2 | 0.046s | 0.031s |

**BẢNG 4. Tối kích thước chương trình** **arm**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Tên chương trình** | **Mô tả chương trình** | **Kỹ thuật tối ưu thực hiện** | **Ct gốc** | **Ct tối ưu** |
| 11 | Tính tổng bình phương các số lẻ từ 1 đến < n |  | Dùng cờ O2 | 1552 bytes | 1446 bytes |
| 12 | Tìm max của 3 số a,b,c nhập từ bàn phím |  | Dùng cờ O2 | 1820 bytes | 1931 bytes |
| 13 | Nhập một số nguyên từ bàn phím, kiểm tra xem đó là số chẵn hay lẻ |  | Dùng cờ O2 | 1088 bytes | 1157 bytes |
| 14 | Tìm ước số chung lớn nhất và bội số chung nhỏ nhất của 2 số nguyên nhập từ bàn phím |  | Dùng cờ O2 | 1957 bytes | 1778 bytes |
| 15 | Nhập một số nguyên từ bàn phím. Kiểm tra một số có phải là số hoàn hảo? |  | Dùng cờ O2 | 2097 bytes | 1936 bytes |
| 16 | Nhập vào dãy số tính tổng các số lẻ trong dãy. |  | Dùng cờ O2 | 3370 bytes | 2625 bytes |
| 17 | Nhập vào 1 mảng và in các số nguyên tố trong mảng và vị trí của nó |  | Dùng cờ O2 | 2568 bytes | 2428 bytes |
| 18 | Nhập vào 1 dãy số và in ra vị trí lớn nhất |  | Dùng cờ O2 | 2615 bytes | 2357 bytes |
| 19 | Nhập vào 1 mảng và xuất mảng đó ra console. |  | Dùng cờ O2 | 4660 bytes | 3588 bytes |
| 20 | Nhập vào 1 mảng và tìm kiếm 1 giá trị bất kỳ trong mảng và in ra. |  | Dùng cờ O2 | 4150 bytes | 3368 bytes |