

# Инструкции

# Инструкции

- \* Основная арифметика
  - \* Битовые операции
  - \* Операции переходов
  - \* Строковые операции
  - \* Стековые операции
  - \* Операция MOV
- \* Используют явно и не явно регистры процессора
  - \* Почти все инструкции влияют на регистр флагов
  - \* В защищенном режиме не все инструкции могут выполняться пользователем

# Инструкции

- \* Основная арифметика
- \* Битовые операции
- \* Операции переходов
- \* Строковые операции
- \* Стековые операции
- \* Операция MOV

- \* Используют явно и не явно регистры процессора
- \* Почти все инструкции влияют на регистр флагов
- \* В защищенном режиме не все инструкции могут выполняться пользователем

## Основная арифметика

ADD, SUB, DIV, MUL, INC, DEC, LEA

## Битовые операции

AND, OR, XOR, SHR, SHL, NOT

## Операции переходов

JMP, Jx, CALL, INT, RET, IRET

## Строковые операции

LODS, STOS, MOVS, INS, OUTS, CMPS  
(+ приставки REP/REPE/REPNE)

# Стековые операции

- \* PUSH/POP
- \* PUSH/POPA
- \* PUSHAD/POPAD
- \* PUSHF/POPF (+D, +Q)

# Операции различных расширений

- \* FPU
- \* MMX
- \* SSE

Расширения архитектуры x86

# Стековые операции

- \* PUSH/POP
- \* PUSH/POPA
- \* PUSHAD/POPAD
- \* PUSHF/POPF (+D, +Q)

# Операции различных расширений

- \* FPU
- \* MMX
- \* SSE

Расширения архитектуры x86

# Операция MOV

Компилируем отныне только mov-ами

# Примеры замены операций mov-ом

## INC

```
.text
mov eax, x
mov eax, [increment+eax]

.data
increment: DB 1, ..., 255, 0
```

## AND

```
.text
mov eax, x
mov edx, y
mov eax, [a0+eax]
mov al, [eax+edx]

.data
a0: DD a10, a11
a10: DB 0, 0
a11: DB 0, 1
```

## CMP

```
mov byte ptr [ax], 0
mov byte ptr [dx], 1
mov al, [ax]
```