

БАЗОВЫЕ ПОНЯТИЯ ПРОГРАММИРОВАНИЯ

*Хорошее начало – половина дела.
Платон*

Основы программирования

Содержание

2

- Информация
- Алгоритм
- Компьютер
- Программа
- Программирование
- Основная проблема программирования

Информация

3

- *Информация* (лат. informatio – разъяснение) – сведения об объектах, явлениях, их параметрах, свойствах, состоянии, которые *уменьшают имеющуюся о них степень неопределенности, неполноты знаний.*
(Н.В. Макарова)
- *Информация* – это обозначение содержания, полученного из внешнего мира в процессе нашего приспособления к нему и приспособления к нему наших чувств.
(Н. Винер)

Информация vs данные

4

- *Данные* – хранимые, но *не используемые* сведения об объектах, явлениях, их параметрах, свойствах, состоянии.



- Информация характеризует не сообщение, а *соотношения между сообщением и его потребителем*. Без наличия потребителя (хотя бы потенциального) говорить об информации бессмысленно.

Измерение информации

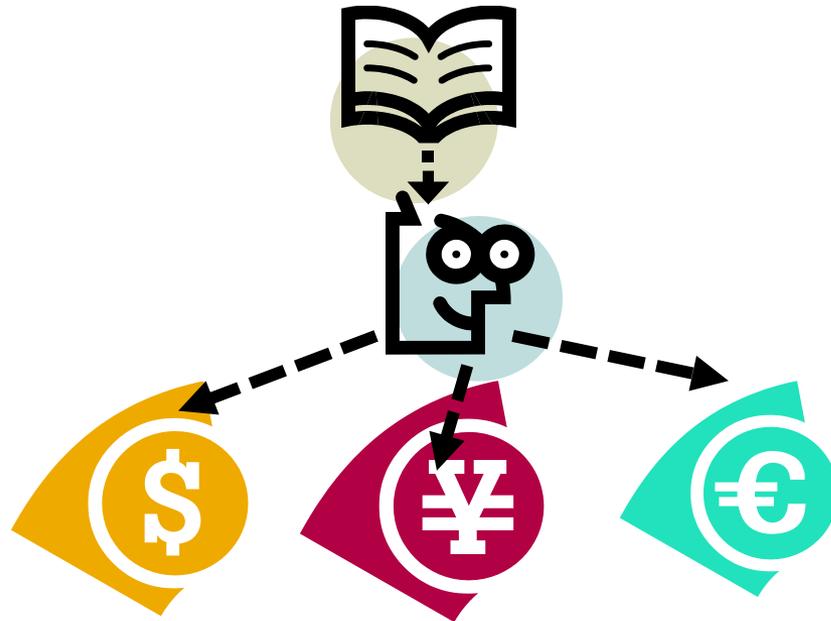
5

- Параметры измерения информации:
 - ▣ Объем данных V
 - ▣ Количество информации I
- Меры информации:
 - ▣ Прагматическая мера
 - ▣ Семантическая мера
 - ▣ Синтаксическая мера

Прагматическая мера

6

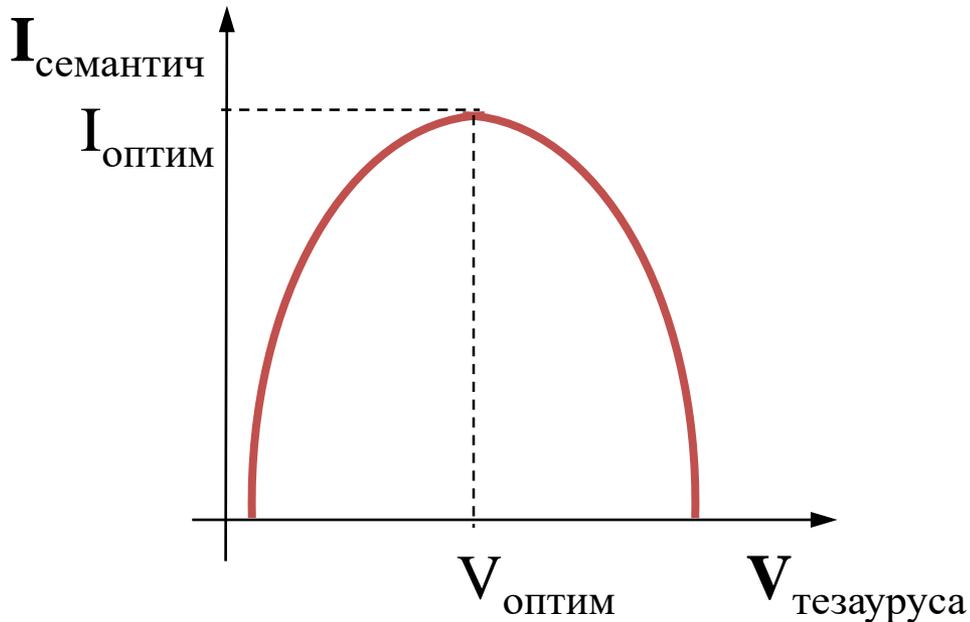
- *Прагматическая мера* используется для измерения полезности и ценности информации для достижения поставленной цели.
- $I_{\text{прагм}}$ измеряется в тех же единицах, что и целевая функция



Семантическая мера

7

- *Семантическая мера* используется для измерения информации относительно имеющегося в ней смысла.
- Относительная мера количества информации $S = I_{\text{семантич}} / V$
- *Тезаурус* – априорные знания пользователя.



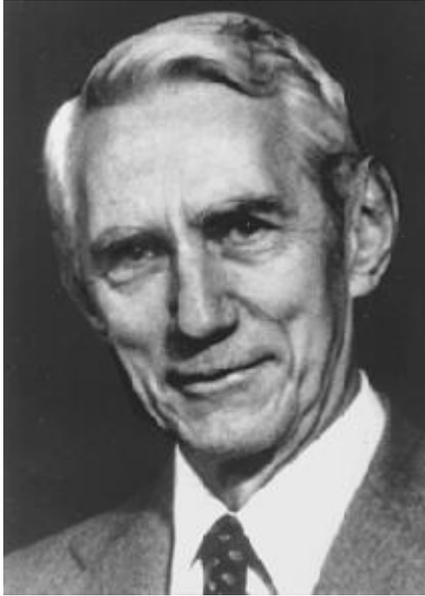
Синтаксическая мера

8

- *Синтаксическая мера* используется для измерения информации безотносительно к объекту.
- V – количество символов в сообщении
- *Подход Р. Хартли*: получение информации есть выбор одного сообщения из конечного наперед заданного множества равновероятных сообщений.
 - $I_{\text{синт}} = \log_2 V$

Синтаксическая мера

9



Клод Эльвуд Шеннон
(1916-2001)

- *Подход К. Шеннона*: получение информации есть выбор одного сообщения из конечного наперед заданного множества сообщений, не обязательно равновероятных.

$$\square \mathbf{I}_{\text{синт}} = - \sum_{i=1}^V p_i \log_2 p_i$$

энтропия сообщения

p_i – вероятность появления i -го символа

Синтаксическая мера

10

- Сообщение:
To be or not to be? That is a question!
- Количество символов:
 $V=39$
- Частоты и вероятности символов:
- Энтропия сообщения:
По Хартли: $I=5,28$
По Шеннону: $I=3,48$

Символ	Частота	Вероятность
A	2	5%
B	2	5%
E	3	8%
H	1	3%
I	2	5%
N	2	5%
O	5	13%
Q	1	3%
R	1	3%
S	2	5%
T	6	15%
U	1	3%
?	1	3%
!	1	3%
—	9	23%
Всего	39	100%

Показатели качества информации

11

- *Адекватность информации* – уровень соответствия реального объекта образу, созданному с помощью полученной информации.
- *Репрезентативность* – правильность отбора и формирования информации для адекватного отражения свойств объекта.
- *Актуальность* – степень сохранения ценности информации с течением времени.
- *Содержательность* – семантическая емкость информации, $I_{\text{семантич}}/V$.
- *Лаконичность* сообщения – величина $I_{\text{синтаксич}}/V$.

Синтаксическая мера

12

□ *Сообщение*
abcdefghij

□ *Количество символов*
 $V=10$

□ *Энтропия сообщения*
 $I=-(10*\log_2 0,1)=33,2$

□ *Лаконичность сообщения*
 $I/V = 3,2$

□ *Сообщение*
aaaaabbbbb

□ *Количество символов*
 $V=10$

□ *Энтропия сообщения*
 $I=-(2*\log_2 0,5)=2$

□ *Лаконичность сообщения*
 $I/V = 0,2$

Единицы измерения информации

13

- *Бит* (*bit*, *binary digit* – двоичная цифра) – количество информации, минимально необходимое для различения двух равновероятных сообщений.
- *Бит* – минимальный объем памяти компьютера, необходимый для хранения числа двоичной системы счисления (0 или 1).
- *Дигит* (*dit*, *decimal digit* – десятичная цифра) – альтернативная единица измерения информации, ... 10 равновероятных сообщений.

Представление информации в компьютере

14

- Для представления в компьютере все символы сообщения *кодируются* с помощью последовательности битов.
- *Бит* – число двоичной системы счисления, которое может принимать значение 0 или 1.
- Для представления символов на клавиатуре компьютера (буквы английского и русского языков, цифры, знаки пунктуации и др.) используется последовательность битов.

Таблица ASCII

- Кодировка символов на клавиатуре компьютера зафиксирована в *таблице ASCII (American Standard Code for Information Interchange, американский стандартный код для обмена информацией)*.

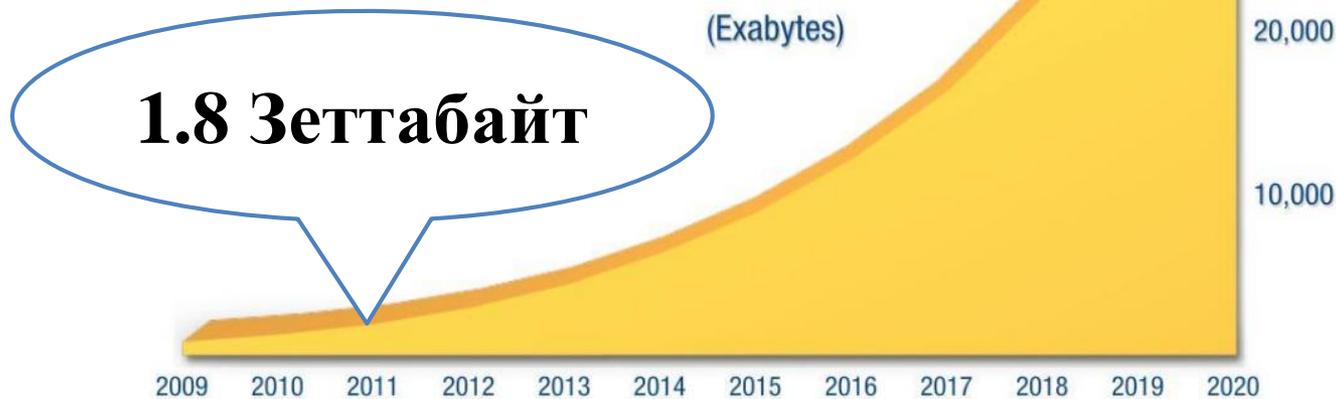
	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	.A	.B	.C	.D	.E	.F
0.		☺	☹	♥	♦	♣	♠	•	◐	◑	♂	♀	♂	♂	♂	♂
1.	▶	◀	↕	!!	¶	§	—	↑	↓	→	←	↵	↔	▲	▼	
2.		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
3.	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4.	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5.	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
6.	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7.	p	q	r	s	t	u	v	w	x	y	z	{		}	~	Δ

	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	.A	.B	.C	.D	.E	.F
8.	A	B	V	Г	Д	Е	Ж	Э	И	Й	К	Л	М	Н	О	П
9.	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
A.	а	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п
B.	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
C.	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
D.	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
E.	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я
F.	Ё	ё	≥	≤	∫	∫	÷	≈	°	•	·	√	π	²	■	

Всемирный информационный потоп

16

- Объем мировых данных увеличится в **50 раз** с **2010 по 2020 гг.**



Единица	Значение
1 байт	8 бит
1 Кбайт	2^{10} байт
1 Мбайт	2^{10} Кбайт
1 Гбайт	2^{10} Мбайт
1 Тбайт	2^{10} Гбайт
1 Пбайт	2^{10} Тбайт
1 Эбайт	2^{10} Пбайт
1 Збайт	2^{10} Эбайт
1 Йбайт	2^{10} Збайт

- Бизнес: электронная коммерция, банковские транзакции, биржи и др.
- Наука: сенсорные сети, данные моделирования и др.
- Общество: новостные ленты, социальные сети, YouTube и др.

Всемирный информационный потоп

17



- 1.8 Зеттабайт – это 57.5 млрд. iPad 32 Гб для постройки "Великой китайской iPad стены"

Алгоритм

18

- *Алгоритм* – точное предписание исполнителю совершить определенную последовательность действий для достижения поставленной цели за конечное число шагов.

Дано:

a, b, c – вещественные

Найти:

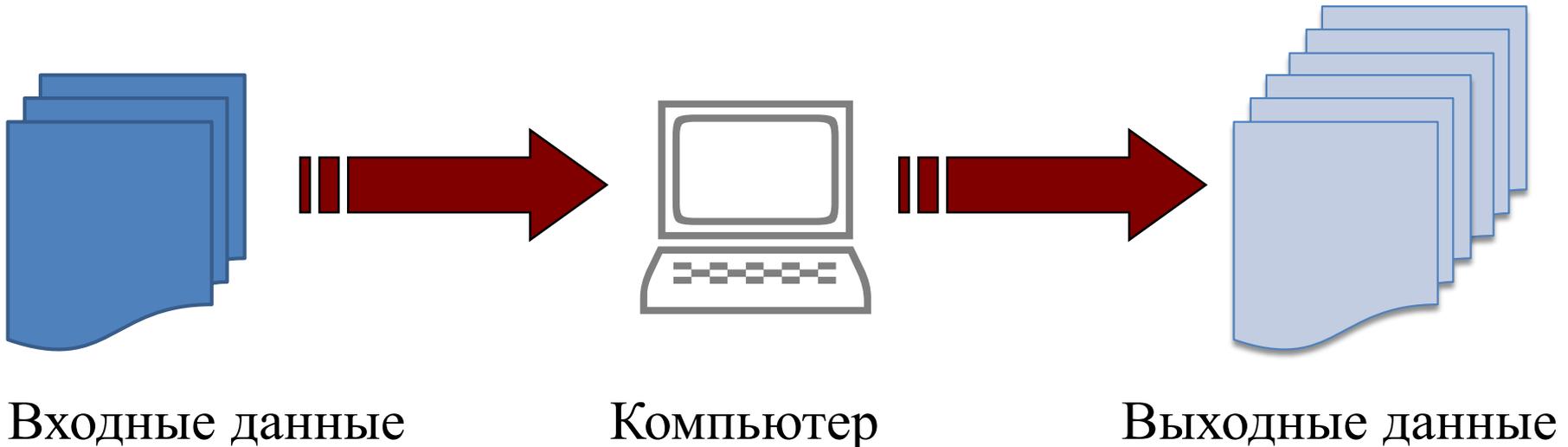
корни уравнения $ax^2+bx+c=0$

```
Алгоритм КОРНИ;  
вход вещ a, b, c;  
выход вещ x1, x2;  
перем вещ D;  
начало  
  D:=b*b-4*a*c;  
  если D<0 то  
    аварвыход ("Корней нет");  
  иначе  
    x1:=(-b+sqrt(D))/(2*a);  
    x2:=(-b-sqrt(D))/(2*a);  
  конец если  
конец
```

Компьютер

19

- *Компьютер* – электронный вычислитель, производящий обработку информации (данных) в соответствии с закладываемыми в него программами.



Программа

20

- *Компьютерная программа* – последовательность команд.
- *Команда* – описание операции, которую должен выполнить компьютер. Команда имеет *код* (условное обозначение), *исходные данные* (операнды) и *результат*.

КОП	АдрОперанд1	АдрОперанд2	АдрРезультат
10111000	00000001	00000000	00000000

Загрузить 1
в ячейку
памяти с
адресом 0

Принципы Фон Неймана

21

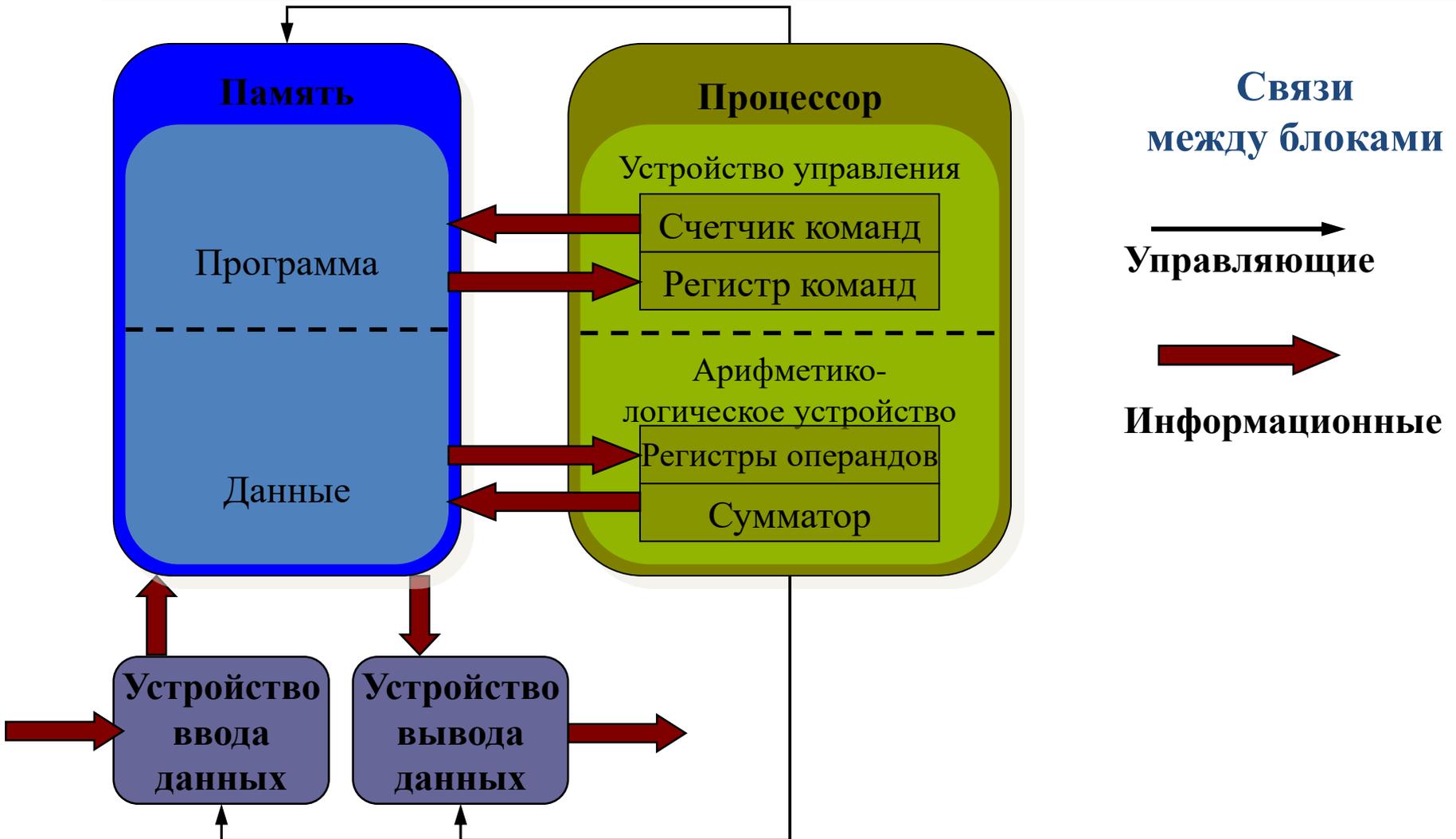


**Джон фон
Нейман**
(1903-1954),
в 1946 г.

- 1. Программное управление*
Программа состоит из набора команд, которые автоматически выполняются процессором (без вмешательства человека).
- 2. Однородность памяти*
Программы и данные хранятся в одной и той же памяти.
- 3. Адресность памяти*
Память состоит из пронумерованных ячеек. Процессору в произвольный момент времени доступна любая ячейка.

Принципы построения компьютера

22



Процессор и память

23

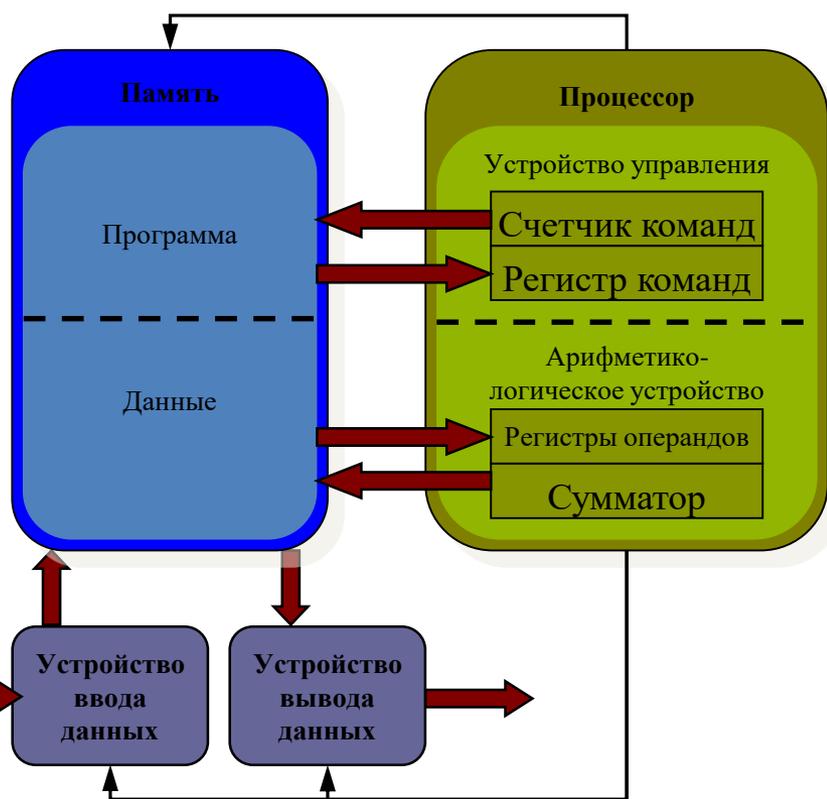
- **Функции процессора:**
 - ▣ обработка данных по заданной программе путем выполнения арифметических и логических операций;
 - ▣ программное управление работой устройств компьютера
- **Функции памяти:**
 - ▣ прием данных из других устройств;
 - ▣ хранение данных;
 - ▣ выдача данных по запросу в другие устройства.

Регистры

- *Регистр* – специализированная ячейка для промежуточного хранения данных.
- *Сумматор* – регистр АЛУ, участвующий в выполнении каждой операции.
- *Счетчик команд* – регистр УУ, который хранит адрес очередной выполняемой команды.
- *Регистр команд* – регистр УУ, который хранит команду в период ее выполнения (код операции и адреса операндов).

Алгоритм работы компьютера

25



0. НАЧАЛО

1. УУ: Выбрать команду по адресу памяти из УУ.СчетчикКоманд
2. УУ: Увеличить УУ.СчетчикКоманд на длину команды
3. УУ: Записать команду в УУ.РегистрКоманд
4. УУ: Если команда="СТОП" то перейти к шагу 10.
5. УУ: Выбрать операнды по адресам памяти из УУ.РегистрКоманд
6. УУ: Записать операнды в АЛУ.РегистрыОперандов
7. АЛУ: Выполнить операцию над операндами
8. АЛУ: Сохранить результат (в памяти/регистре/...)
9. Перейти к шагу 1.
10. КОНЕЦ

Software, Hardware, Brainware

26

- *Software (программные средства)* – нематериальные компоненты программирования (операционные системы, системы программирования и др.).
- *Hardware (аппаратные средства)* – материальные компоненты программирования (компьютеры, устройства хранения информации и др.).
- *Brainware (алгоритмические средства)* – алгоритмы, структуры данных, языки программирования.

Программирование

27

- *Программирование* – процесс составления упорядоченной последовательности действий (*программы*) для компьютера.

Большая советская энциклопедия.

- *Программирование* – научная дисциплина, изучающая программы для компьютеров и способы их составления, проверки и улучшения.

Большая советская энциклопедия.

Программирование: синонимы

28

- *Компьютерные науки (Computer Science)* изучают структуру, общие свойства, закономерности и методы создания, хранения, поиска, обработки и применения информации в различных сферах человеческой деятельности на базе компьютерной техники.
- *Информатика* (англ. Informatics, фр. Informatique) – аббревиатура от "Информация + Автоматика"
- *Вычислительные науки* – рус., устар.

Программирование: виды

29

- *Прикладное* – разработка программ для конечных пользователей (не программистов).
 - Бухгалтерские программы, системы управления предприятиями, игры, обучающие системы ...
- *Системное* – разработка программ для программистов.
 - ОС и офисные пакеты, СУБД, CASE, ...
- *Научно-исследовательское* – разработка прототипов прикладного/системного ПО для проверки новых алгоритмов и методов обработки данных.

Программирование: люди

30

- *Пользователи* – потребители программ.
- *Программисты* – разработчики программ.
 - ▣ Аналитики, спецификаторы
 - ▣ Кодировщики
 - ▣ Тестировщики
 - ▣ Библиотекари
 - ▣ ...
- *Системные администраторы, инженеры* – обеспечивают техническую сторону разработки и потребления программ.

Программирование в малом vs программирование в большом

31

- *Программирование в малом (programming in the small)* – программирование на уровне операторов небольшой программы, которая может быть разработана одним человеком.
- *Программирование в большом (programming in the large)* – программирование на уровне подсистем (библиотек) большой программы, которая не может быть разработана одним человеком.

Промышленное программирование

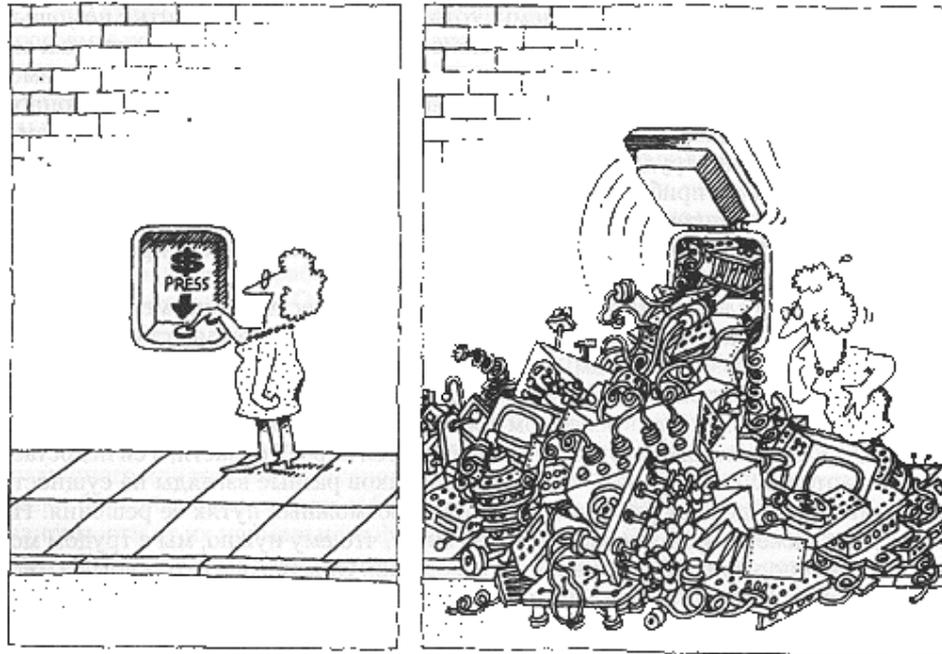
32

- Промышленно создаваемое ПО имеет
 - большое время жизни
 - большое количество пользователей, зависящих от его нормального функционирования
 - большую команду разработчиков.

Программирование: основная проблема

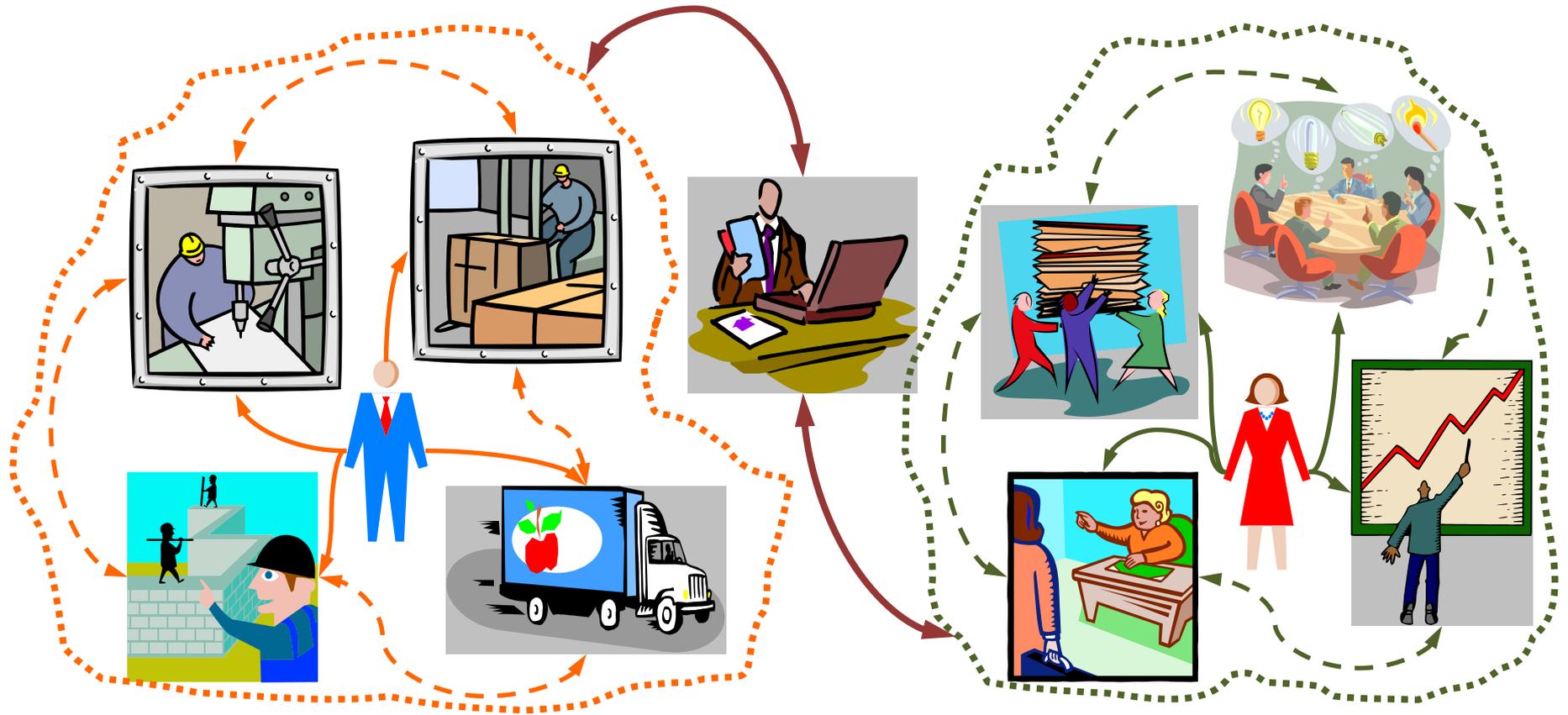
33

СЛОЖНОСТЬ



Сложность предметной области

34



Сложность определения требований к программе

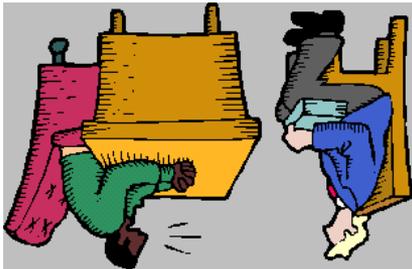
35



- Пользователю сложно объяснить разработчикам, *что* должна делать система (часто он сам этого не знает).
- Разработчикам сложно *однозначно* зафиксировать требования пользователя.



- Изменение требований пользователя к системе *в процессе* разработки.

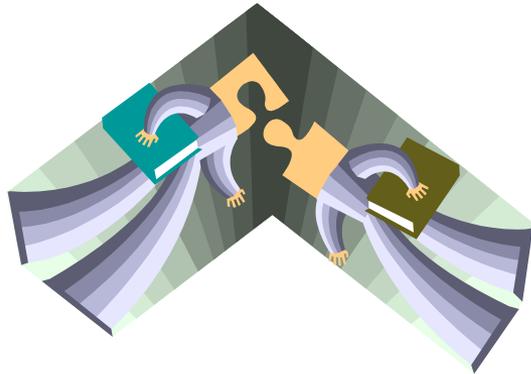


Сложность управления процессом разработки

36



- Невозможность полного понимания *одним* человеком программной системы объемом в десятки-сотни тысяч строк.
- Сложность координации между разработчиками.



Сложность поиска и исправления ошибок

37



- Поиск и исправление ошибок – трудно формализуемая интеллектуальная деятельность.
- Программист подсознательно против того, что в его программе может быть ошибка.

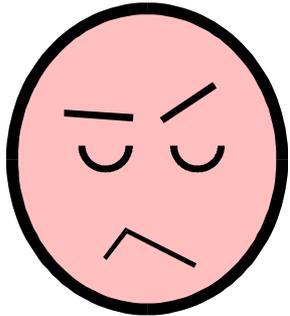


Сложность сопровождения

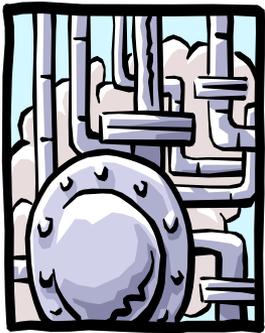
38



- ❑ Устранение ошибок, обнаруженных во время эксплуатации, модернизация системы вследствие изменения нужд пользователей.



- ❑ Сопровождение системы для программиста психологически более трудно, чем собственно разработка системы.



- ❑ Версионирование исходных текстов: сохранение истории изменений с возможностью отката до любого шага.

Средства борьбы со сложностью

39



- Избавиться от сложности нельзя, но с ней можно справиться!
- Технология программирования – комплекс средств для борьбы со сложностью



- *концептуальные средства*, определяющие стиль и методы разработки программ
- *организационные средства*, определяющие форму труда в команде программистов
- *программные средства* разработки.

Заключение

- Информация получается в результате обработки данных.
- Обработка данных производится компьютером в соответствии с закладываемыми в него программами.
- Программа реализует некоторый алгоритм.
- Программирование – научная дисциплина, изучающая способы составления, проверки и улучшения программ.
- Сложность – основная проблема программирования.