NAME

azel − satellite predictions

SYNOPSIS

/usr/games/azel [**−d**] [**−l**] satellite1 [**−d**] [**−l**] satellite2 ...

DESCRIPTION

Azel predicts, in convenient form, the apparent trajectories of Earth satellites whose orbital elements are given in the argument files. If a given satellite name cannot be read, an attempt is made to find it in a directory of satellites maintained by the program's author. The**−d** option causes *azel* to ask for a date and read line 1 data (see below) from the standard input. The **−l** option causes *azel* to ask for the observer's latitude, west-longitude, and height above sea level.

For each satellite given the program types its full name, the date, and a sequence of lines each containing a time, an azimuth, an elevation, a distance, and a visual magnitude. Each such line indicates that: at the indicated time, the satellite may be seen from Murray Hill (or provided location) at the indicated azimuth and elevation, and that its distance and apparent magnitude are as given. Predictions are printed only when the sky is dark (sun more than 5 degrees below the horizon) and when the satellite is not eclipsed by the earth's shadow. Satellites which have not been seen and verified will not have had their visual magnitude level set correctly. All times input and output by*azel* are GMT (Uni versal Time). The satellites for which elements are maintained are:

sla,b,e,f,k Skylab A through Skylab K. Skylab A is the laboratory; B was the rocket but it has crashed. A and probably K have been verified.

cop Copernicus I. Never verified.

oao Orbiting Astronomical Observatory. Seen and verified.

pag Pageos I. Seen and verified; fairly dim (typically 2nd-3rd magnitude), but elements are extremely accurate.

exp19 Explorer 19; seen and verified, but quite dim (4th-5th magnitude) and fast-moving.

c103b, c156b, c184b, c206b, c220b, c461b, c500b
Various of the USSR Cosmos series; none seen.

7276a Unnamed (satellite # 72-76A); not seen.

The element files used by *azel* contain 5 lines. The first line gives a year, month, day, hour, and minute at which the program begins its consideration of the satellite, followed by a number of minutes and an interval in minutes. If the year, month, and day are 0, they are taken to be the current date (taken to change at 6 A.M. local time). The output report starts at the indicated epoch and prints the position of the satellite for the indicated number of minutes at times separated by the indicated interval. This line is ended by 2 numbers that specify options to the program governing the completeness of the report; they are ordinarily both '1'; the first suppresses output when the sky is not dark; the second suppresses output when the satellite is eclipsed by the earth's shadow. The next line of an element file is the full name of the satellite. The next 3 are the elements themselves (including certain derivatives of the elements).

FILES

/usr/jfo/* − orbital element files

SEE ALSO

sky(VI)

NAME

bio − biorhythm analysis

SYNOPSIS

/usr/games/bio birth-date start-date [number-of-days]

DESCRIPTION

Bio produces a graph of a person's biorhythm functions. The date arguments are given in the form mm/dd/yy. The *number-of-days* argument is the number of days for which the graph is printed. The default is 30 days. The three biorhythm curves are plotted: physical (p), emotional (e), and intellectual (i).

To get a neatly formatted graph quickly use:

bio birth-date start-date ^ reform ^ pr ^ gsi

NAME

bj − the game of black jack

SYNOPSIS

/usr/games/bj

DESCRIPTION

Bj is a serious attempt at simulating the dealer in the game of black jack (or twenty-one) as might be found in Reno. The following rules apply:

The bet is $2 every hand.

A player ‘natural’ (black jack) pays $3. A dealer natural loses $2. Both dealer and player naturals is a ‘push’ (no money exchange).

If the dealer has an ace up, the player is allowed to make an ‘insurance’ bet against the chance of a dealer natural. If this bet is not taken, play resumes as normal. If the bet is taken, it is a side bet where the player wins $2 if the dealer has a natural and loses $1 if the dealer does not.

If the player is dealt two cards of the same value, he is allowed to ‘double’. He is allowed to play two hands, each with one of these cards. (The bet is doubled also; $2 on each hand.)

If a dealt hand has a total of ten or eleven, the player may ‘double down’. He may double the bet ($2 to $4) and receive exactly one more card on that hand.

Under normal play, the player may ‘hit’ (draw a card) as long as his total is not over twenty-one. If the player ‘busts’ (goes over twenty-one), the dealer wins the bet.

When the player ‘stands’ (decides not to hit), the dealer hits until he attains a total of seventeen or more. If the dealer busts, the player wins the bet.

If both player and dealer stand, the one with the largest total wins. A tie is a push.

The machine deals and keeps score. The following questions will be asked at appropriate times. Each question is answered by **y** followed by a new line for ‘yes’, or just new line for ‘no’.

? (means, ‘‘do you want a hit?’’)
Insurance?
Double down?

Every time the deck is shuffled, the dealer so states and the ‘action’ (total bet) and ‘standing’ (total won or lost) is printed. To exit, hit the interrupt key (DEL) and the action and standing will be printed.

NAME

chess − the game of chess

SYNOPSIS

/usr/games/chess

DESCRIPTION

Chess is a computer program that plays class D chess. Moves may be given either in standard (descriptive) notation or in algebraic notation. The symbol '+' is used to specify check; 'o-o' and 'o-o-o' specify castling. To play black, type 'first'; to print the board, type an empty line.

Each move is echoed in the appropriate notation followed by the program's reply.

FILES

/usr/lib/book opening 'book'

DIAGNOSTICS

The most cryptic diagnostic is 'eh?' which means that the input was syntactically incorrect.

WARNING

Over-use of this program will cause it to go away.

BUGS

Pawns may be promoted only to queens.

NAME

cubic − three dimensional tic-tac-toe

SYNOPSIS

/usr/games/cubic

DESCRIPTION

Cubic plays the game of three dimensional 4×4×4 tic-tac-toe. Moves are given by the three digits (each 1-4) specifying the coordinate of the square to be played.

WARNING

Too much playing of the game will cause it to disappear.

NAME

factor − discover prime factors of a number

SYNOPSIS

/usr/games/factor

DESCRIPTION

When *factor* is invoked, it prompts for a number to be typed in. If you type in a positive number less than 2^{56} (about 7.2×10^{16}) it will factor the number and print its prime factors; each one is printed the proper number of times. Then it waits for another number. It exits if it encounters a zero or any non-numeric character.

Maximum time to factor is proportional to $\sqrt{n}$ and occurs when n is prime or the square of a prime. It takes 1 minute to factor a prime near 10^{13}.

DIAGNOSTICS

‘‘Ouch.’’ for input out of range or for garbage input.

NAME

moo − guessing game

SYNOPSIS

/usr/games/moo

DESCRIPTION

Moo is a guessing game imported from England. The computer picks a number consisting of four distinct decimal digits. The player guesses four distinct digits being scored on each guess. A ‘cow’ is a correct digit in an incorrect position. A ‘bull’ is a correct digit in a correct position. The game continues until the player guesses the number (a score of four bulls).

NAME

othello − a game of dramatic reversals

SYNOPSIS

/usr/games/othello [[**−r**] *file*]

DESCRIPTION

Othello (a.k.a *reversi*) is played on an 8 by 8 board using two-sided tokens. Each player takes his turn by placing a token with his side up in an empty square. During the first four turns, players may only place tokens in the four central squares of the board. Subsequently, with each turn, a player *must* capture one or more of his opponents tokens. He does this by placing one of his tokens such that he outflanks one or more of his opponents', horizontally, vertically or diagonally. Captured tokens are flipped over and thus can be re-captured. If a player cannot outflank his opponent he must forfeit his turn. The play continues until the board is filled or until no more outflanking is possible.

In this game, your tokens are asterisks and the machines' are at-signs. You move by typing in the row and column at which you want to place your token as two digits (1-8), optionally separated by blanks or tabs. You can also type in

- **c** to continue the game after hitting break (this is only necessary if you interrupt the machine while it is deliberating),
- **g** *n* to start *othello* playing against itself for the next *n* moves (or until the break key is hit),
- **n** to stop printing the board after each move,
- **o** to start it up again,
- **p** to print the board regardless,
- **q** to quit (without dishonor),
- **s** to print the score, and, as always,
- **!** to escape to the shell. Control-D gets you back.

Othello also recognizes several commands which are valid only at the start of the game, before any moves have been made. They are

- **f** to let the machine go first.
- **h** *n* to ask for a handicap of from one to four corner squares. If you're *really* good, you can give the machine a handicap by typing a negative number.
- **l** *n* to set the amount of lookahead used by the machine in searching for moves. Zero means none at all. Four is the default. Greater than six means you may fall asleep waiting for the machine to move.
- **t** *n* to tell *othello* that you will only need *n* seconds to consider each move. If you fail to respond in the allotted time, you forfeit your turn.

If *othello* is given a file name as an argument, it will checkpoint the game, move by move, by dumping the board onto *file.* The **−r** flag will cause *othello* to restart the game from *file* and continue logging.

DIAGNOSTICS

Illegal! and Huh?

NAME

sky − obtain ephemerides

SYNOPSIS

/usr/games/sky [**−l**]

DESCRIPTION

Sky predicts the apparent locations of the Sun, the Moon, the planets out to Saturn, stars of magnitude at least 2.5, and certain other celestial objects. *Sk y* reads the standard input to obtain a GMT time typed on one line with blanks separating year, month number, day, hour, and minute; if the year is missing the current year is used. If a blank line is typed the current time is used. The program prints the azimuth, elevation, and magnitude of objects which are above the horizon at the ephemeris location of Murray Hill at the indicated time. The**−l** flag causes it to ask for another location.

Placing a ‘‘1’’ input after the minute entry causes the program to print out the Greenwich Sidereal Time at the indicated moment and to print for each body its topographic right ascension and declination as well as its azimuth and elevation. Also, instead of the magnitude, the semidiameter of the body, in seconds of arc, is reported.

A ‘‘2’’ after the minute entry makes the coordinate system geocentric.

The effects of atmospheric extinction on magnitudes are not included; the brightest magnitudes of variable stars are marked with ‘‘*’’.

For all bodies, the program takes into account precession and nutation of the equinox, annual (but not diurnal) aberration, diurnal parallax, and the proper motion of stars. In no case is refraction included.

The program takes into account perturbations of the Earth due to the Moon, Venus, Mars, and Jupiter. The expected accuracies are: for the Sun and other stellar bodies a few tenths of seconds of arc; for the Moon (on which particular care is lavished) likewise a few tenths of seconds. For the Sun, Moon and stars the accuracy is sufficient to predict the circumstances of eclipses and occultations to within a few seconds of time. The planets may be off by several minutes of arc.

There are lots of special options not described here, which do things like substituting named star catalogs, smoothing nutation and aberration to aid generation of mean places of stars, and making conventional adjustments to the Moon to improve eclipse predictions.

For the most accurate use of the program it is necessary to know that it actually runs in Ephemeris time.

FILES

/usr/lib/startab, /usr/lib/moontab

SEE ALSO

azel (VI)

American Ephemeris and Nautical Almanac, for the appropriate years; also, the *Explanatory Supplement to the American Ephemeris and Nautical Almanac.*

NAME

ttt − the game of tic-tac-toe

SYNOPSIS

/usr/games/ttt

DESCRIPTION

Ttt is the X and O game popular in the first grade. This is a learning program that never makes the same mistake twice.

Although it learns, it learns slowly. It must lose nearly 80 games to completely know the game.

FILES

/usr/games/ttt.k learning file

NAME

wump − the game of hunt-the-wumpus

SYNOPSIS

/usr/games/wump

DESCRIPTION

Wump plays the game of ‘‘Hunt the Wumpus.’’ A Wumpus is a creature that lives in a cave with several rooms connected by tunnels. You wander among the rooms, trying to shoot the Wumpus with an arrow, meanwhile avoiding being eaten by the Wumpus and falling into Bottomless Pits. There are also Super Bats which are likely to pick you up and drop you in some random room.

The program asks various questions which you answer one per line; it will give a more detailed description if you want.

This program is based on one described in *People’s Computer Company, 2,* 2 (November 1973).

BUGS

It will never replace Space War.