```
# JuG on Mon Oct 27 09:45:11 2025
# Flow-chart etude NIDA

# Libraries
import polars as pl

# Read data
dm01 = pl.read_csv("./ascii-data-files_nida-ctn-0001/dm.csv")
dm02 = pl.read_csv("./ascii-data-files_nida-ctn-0002/dm.csv")

print(dm01["ARM"].value_counts())
print(dm02["ARM"].value_counts())

dm = pl.concat([dm01, dm02])

# Verify unique patients
print((dm.group_by("USUBJID").len().filter(pl.col("len") > 1).height))

# Description of number of randomized patients
print(dm["ARM"].value_counts())

dmIncluded = dm.filter(pl.col("ARM") != "SCREEN FAILURE")
print(dmIncluded["ARM"].value_counts())

# Discharge
ds01 = pl.read_csv("./ascii-data-files_nida-ctn-0001/ds.csv")
ds02 = pl.read_csv("./ascii-data-files_nida-ctn-0002/ds.csv")

ds = pl.concat([ds01, ds02])

dsDischarged = (ds
    .filter(pl.col("EPOCH") != "SCREENING")
    .with_columns((pl.col("VISITNUM") < 14).alias("beforeEndActive"))
    .filter(
        pl.col("beforeEndActive") &
        (pl.col("DSDECOD") != "PARTICIPANT COMPLETED ACTIVE PHASE OF STUDY")
    )
    .select(["STUDYID", "USUBJID", "DSDECOD", "VISITNUM"])
    .unique()
)

print(dsDischarged["VISITNUM"].value_counts())
print(dsDischarged["DSDECOD"].value_counts())

dtf = dmIncluded.join(dsDischarged, on="USUBJID", how="left")

# For cross-tabulation, you can use pivot or group_by
visitnum_counts = (dtf
    .group_by(["ARMCD", "VISITNUM_right"])
    .len()
    .pivot(index="ARMCD", columns="VISITNUM_right", values="len")
)
print(visitnum_counts)

dsdecod_counts = (dtf
    .group_by(["DSDECOD", "ARMCD"])
    .len()
    .pivot(index="DSDECOD", columns="ARMCD", values="len")
)
print(dsdecod_counts)
```